

Konzeption, Implementierung und betriebswirtschaftliche Analyse eines
prototypischen Systems autonomer, dezentral organisierter
Software-Agenten zur Durchführung zweiseitiger kombinatorischer
Auktionen von multimodalen Transportdienstleistungen
am Beispiel des kombinierten Verkehrs

Dissertation

zur Erlangung des Doktorgrades
Dr. rer. pol.

der Fakultät für Wirtschaftswissenschaften
der Universität Duisburg-Essen

vorgelegt
von

Dipl.-Kfm. René Föhring

aus

Bochum

Tag der mündlichen Prüfung: 6. Juli 2017

Gutachter:

Univ.-Prof. Dr. Stephan Zelewski
Univ.-Prof. Dr. Tobias Kollmann
PD Dr. Christina Klüver

Danksagung

Diese Forschungsarbeit ist das Resultat des konsequent innovativen Weiterdenkens meiner Arbeiten am EU-Projekt „CODE24“. Sie ist ebenso das Ergebnis vieler Nächte und Wochenenden – vertieft in Bildschirme, Bücher und Quelltext.

Ich muss zuvorderst meinem Doktorvater, Univ.-Prof. Dr. Stephan Zelewski, danken, der in mir das Potential gesehen hat, diese Arbeit zu verfassen. Mein Interesse an der wissenschaftlichen Praxis wäre ohne das Vorbild seines streng gelebten, aber für modernste Entwicklungen stets offenen, Wissenschaftsbildes nie geweckt worden. Danke!

In besonderem Maße muss ich mich zudem beim PIM-Institut und seinen Mitarbeitern bedanken. Ich blicke auf viele Jahre zurück, in denen ich mich am Institut immer willkommen, herausgefordert und gehört fühlte – egal, ob als Student, wissenschaftliche Hilfskraft, Doktorand oder Lehrbeauftragter. Mir ist bewusst, dass dies nicht selbstverständlich ist und ich bin sehr dankbar für diese Erfahrung.

Zu guter Letzt kann ich mit dem Abschluss dieser Arbeit nun endlich all jenen Menschen aus meinem privaten Umfeld danken, die mich unermüdlich gefördert, motiviert und getragen haben. Habt vielen Dank! Ich liebe euch alle sehr.

Inhaltsverzeichnis

Abkürzungs- und Akronymverzeichnis	II
Symbolverzeichnis	VII
Abbildungsverzeichnis	VIII
Tabellenverzeichnis	IX
Listingverzeichnis	X
1 Einleitung	1
1.1 Forschungsmotivation und Zielsetzung	1
1.2 Forschungsstand	3
1.3 Aufbau der Forschungsarbeit	4
2 Rahmenbedingungen für die Etablierung einer europäischen Online-Frachtenbörse	6
2.1 Hintergrund am Beispiel der Entwicklung des Schienengüterverkehrs	6
2.2 Realprobleme im Schienengüterverkehr	9
2.3 Das Projekt CODE24	12
2.4 Erkenntnisse zu Online-Frachtenbörsen im Rahmen des Projekts CODE24	14
2.4.1 Ziele der Etablierung	14
2.4.2 Anforderungen potentieller Nutzer	17
2.4.2.1 Prozesse und Sicherheit	17
2.4.2.2 Technologie	20
2.4.2.3 Leistungen	21
2.4.2.4 Betreiber	27
2.4.2.5 Finanzen	29
2.4.3 Analyse des kompetitiven Umfelds	30
2.4.4 Branchenanalyse	33
2.4.5 Abschätzung des Marktpotentials	38
2.4.6 Vorliegende Implementierung eines Softwareprototyps	44
2.4.6.1 Ziele	44
2.4.6.2 Leistungsumfang	45
2.4.7 Kritische Reflexion der bisherigen Entwicklung	50
3 Betriebswirtschaftliche Desiderata im Schienengüterverkehr	55
4 Wissenschaftliche Probleme	59
5 Begrifflichkeiten und konzeptionelle Grundlagen	60
5.1 Elektronische Marktplätze	60
5.1.1 Marktplatzbegriff	60
5.1.2 Der Begriff des elektronischen Marktplatzes	61
5.1.3 Merkmale von elektronischen Marktplätzen	62
5.1.4 Arten von elektronischen Marktplätzen	63
5.1.4.1 Vertikale und horizontale Marktplätze	63
5.1.4.2 Käufer- und verkäuferorientierte sowie neutrale Marktplätze	63
5.1.4.3 Offene und geschlossene Marktplätze	67
5.1.4.4 Marktplätze mit und ohne Preismechanismus	67

5.1.5	Anforderungen an elektronische Marktplätze	68
5.1.6	Voraussetzungen für den Aufbau von elektronischen Marktplätzen .	71
5.1.7	Geschäftsmodelle	72
5.1.8	Relevanz für die wissenschaftlichen Probleme	73
5.2	Multi-Agenten-Systeme	74
5.2.1	Einführung	74
5.2.2	Begriff des Multi-Agenten-Systems	74
5.2.3	Begriff des Agenten	75
5.2.4	Eigenschaften von Agenten	81
5.2.5	Umgebungen	83
5.2.6	Konzeptionelle Modelle	86
5.2.7	Koordination in Multi-Agenten-Systemen	90
5.2.8	Implementierung agentenbasierter Softwaresysteme	92
5.2.9	Relevanz für die wissenschaftlichen Probleme	93
5.3	Auktionen	94
5.3.1	Einführung	94
5.3.2	Klassische Auktionsformen	98
5.3.3	Moderne Auktionsformen	99
5.3.3.1	Begriff der mehrdimensionalen Auktion	99
5.3.3.2	Begriff der kombinatorischen Auktion	100
5.3.3.3	Ziele des Einsatzes kombinatorischer Auktionen	102
5.3.3.4	Probleme von kombinatorischen Auktionen	103
5.3.3.5	Modell einer kombinatorischen Auktion	107
5.3.3.6	Modell einer Börse	108
5.3.4	Relevanz für die wissenschaftlichen Probleme	112
5.4	Transfer der Grundlagen	112
6	Konzeption eines dezentralen agentenorientierten elektronischen Markt-	
	platzes	115
6.1	Anforderungen	115
6.2	Multi-Agenten-Systeme als dezentrale Marktplätze	116
6.2.1	Herausforderungen der Dezentralisierung	116
6.2.2	Grundlagen der Kontakthanbahnung	119
6.2.3	Funkfeuer-Agenten	119
6.2.4	Rundruf-Kanäle	121
6.2.5	Relevanz für die prototypische Implementierung	122
6.3	Verkehrsnetz	122
6.3.1	Konzeptionelle Grundlagen	122
6.3.2	Haltepunkte	124
6.3.3	Verbindungen und Relationen	125
6.3.4	Anwendungsbeispiel	131
6.3.5	Relevanz für die prototypische Implementierung	133
6.4	Agentenpräferenzen	133
6.4.1	Begriff und Bedeutung der Präferenzen	133
6.4.2	Struktureller Aufbau	134
6.4.3	Möglichkeit der Kompatibilitätsprüfung	137
6.4.4	Relevanz für die prototypische Implementierung	137
6.5	Gruppenbildung	138
6.5.1	Notwendigkeit der Gruppenbildung im dezentralen Netzwerk	138
6.5.2	Vorgehensweise	141
6.5.2.1	Gründung von Gruppen	141

6.5.2.2	Einladungen zu existierenden Gruppen	142
6.5.2.3	Zusammenschluss von existierenden Gruppen	142
6.5.2.4	Vorschläge für den Zusammenschluss von Gruppen durch Mitglieder	145
6.5.3	Relevanz für die prototypische Implementierung	145
6.6	Ad-hoc-Auktionen	146
6.6.1	Begriff der Ad-hoc-Auktion	146
6.6.2	Begriff der Vollständigkeit von Gruppen	147
6.6.3	Anbahnung von Ad-hoc-Auktionen	148
6.6.4	Gruppenphasen im Rahmen von Ad-hoc-Auktionen	149
6.6.4.1	Überblick	149
6.6.4.2	INIT-Phase	150
6.6.4.3	PRE_AUCTION-Phase	151
6.6.4.4	IN_AUCTION-Phase	151
6.6.4.5	POST_AUCTION-Phase	151
6.6.4.6	FINAL-Phase	151
6.6.4.7	MERGED-Phase	152
6.6.5	Transportdienstleistungen als Güter in Auktionen	152
6.6.6	Relevanz für die prototypische Implementierung	153
6.7	Zusammenführung der Erkenntnisse	154
7	Implementierung einer agentenorientierten Online-Frachtenbörse	155
7.1	Ziele der prototypischen Implementierung	155
7.2	Implementierungstechniken	155
7.2.1	In Betracht gezogene Implementierungstechniken	155
7.2.2	Kriterien zur Bewertung der Implementierungstechniken	157
7.2.3	Beschreibung der Implementierungstechniken	159
7.2.3.1	Agentenprogrammiersprache	159
7.2.3.2	Datenformat	164
7.2.3.3	Datenbanksystem	171
7.2.3.4	Webframework	175
7.2.3.5	Webserver	177
7.2.3.6	Versionsverwaltung	178
7.2.3.7	Browsertechnologien	180
7.2.4	Auswahl von Implementierungstechniken	182
7.2.4.1	Vorbemerkungen	182
7.2.4.2	Agentenprogrammiersprache	183
7.2.4.3	Datenformat	184
7.2.4.4	Datenbanksystem	185
7.2.4.5	Webframework	186
7.2.4.6	Webserver	188
7.2.4.7	Versionsverwaltung	188
7.3	Implementierung der integrierten Entwicklungsumgebung	190
7.3.1	Motivation	190
7.3.2	Ziele der Implementierung der Entwicklungsumgebung	191
7.3.3	Leistungsumfang	191
7.3.3.1	Benutzeroberfläche	191
7.3.3.2	Persistenz	193
7.3.3.3	Projekte	193
7.3.3.4	Verkehrsnetz	194
7.3.3.5	Agentenkonfigurationen	201

7.3.3.6	Szenarien	209
7.3.3.7	Simulationen	211
7.4	Implementierung der Agenten-Software	215
7.4.1	Grundlagen	215
7.4.2	Schnittstelle der IDE zur Agenten-Software	216
7.4.2.1	Notwendigkeit von Startparametern	216
7.4.2.2	Notation	217
7.4.2.3	Beispiel	219
7.4.3	Leistungsumfang	223
7.4.3.1	Kommunikationsmechanismen	223
7.4.3.2	Kontaktweiterleitung	223
7.4.3.3	Präferenzoffenlegung und -weiterleitung	224
7.4.3.4	Implementiertes Auktionsmodell	225
7.4.3.5	Durchführung von Ad-hoc-Auktionen	227
7.4.3.6	Verarbeitung der Auktionsergebnisse im Netzwerk	228
8	Installation und Betrieb des Softwareprototyps	229
8.1	Vorbemerkungen	229
8.2	Installation des Softwareprototyps	230
8.3	Betrieb des Softwareprototyps	230
9	Simulation des Multi-Agenten-Systems	232
9.1	Aufbau der Szenarien	232
9.2	Szenario: Spezialfall 1	232
9.3	Szenario: Spezialfall 2	234
9.4	Szenario: Spezialfall 3	235
9.5	Szenario: Spezialfall 4	237
10	Diskussion und Fazit	242
10.1	Diskussion der Ergebnisse	242
10.2	Weiterentwicklung der prototypischen Implementierung	244
10.2.1	Einordnung der Weiterentwicklungspotentiale	244
10.2.2	Weiterentwicklungspotentiale aus betriebswirtschaftlicher Sicht	244
10.2.2.1	Diskussion der vorliegenden Ergebnisse	244
10.2.2.2	Tests der Software mit Akteuren aus relevanten Branchen	245
10.2.2.3	Auktionsmodell	245
10.2.2.4	Beschränkung auf trusted peers	245
10.2.2.5	Beschränkung auf web of trust	246
10.2.2.6	Zertifikate	246
10.2.2.7	Politisch-rechtliche Potentiale	247
10.2.3	Weiterentwicklungspotentiale aus technologischer Sicht	247
10.2.3.1	Benutzeroberfläche der IDE	247
10.2.3.2	Entwurf einer agenten-individuellen Benutzeroberfläche	248
10.2.3.3	Verschlüsselung der Datenübertragungen	249
10.2.3.4	Agentenpräferenzen	249
10.2.3.5	Öffnung interner Datenformate für Drittanbieter	249
10.2.3.6	Algorithmen und LP-Solver	250
10.2.3.7	Signierte Auktionsergebnisse	250
10.3	Weitere Anwendungsgebiete der entwickelten Technologie	250
10.3.1	Decision-Support-Systeme	250
10.3.2	Kontaktanbahnung zwischen potentiellen Geschäftspartnern	251
10.4	Ausblick auf weitere Forschungsperspektiven	251

10.5 Fazit	252
Literaturverzeichnis	253
Anhang	274
A1 – Dokumentation der Datenbanktabellen der IDE	274
A2 – Dokumentation des Quelltextes der Software-Agenten	282

Abkürzungs- und Akronymverzeichnis

ACM	Association for Computing Machinery
AFEX	Agent-Based Freight Exchange
AG	Aktiengesellschaft
AND	Automotive Navigation Data
API	Application Programming Interface
ARL	Akademie für Raumforschung und Landesplanung
ASP	Application Service Provider
B2B	Business to Business
B2C	Business to Consumer
BDI	Belief - Desire - Intentions
bspw.	beispielsweise
BWL	Betriebswirtschaftslehre
bzw.	beziehungsweise
ca.	circa
CLI	Command Line Interface
CLR	Common Language Runtime
COBOL	Common Business Oriented Language
CODE24	Corridor Development 24
CSS	Cascading Style Sheets
CVS	Concurrent Versions System
DB	Deutsche Bahn
DIUM	distancier international uniforme marchandises
DOM	Document Object Model
DSL	Domain Specific Language
DTD	Document Type Definition
EC	European Commission
ECE	Economic Commission for Europe
ECMA	Ecma International (früher: European Computer Manufacturers Association)

ECMT	European Conference of Ministers of Transport
e.g.	exempli gratia
ERP	Enterprise Resource Planning
et al.	et alii
EU	Europäische Union
EUR	Euro
EVU	Eisenbahnverkehrsunternehmen
FIPA	Foundation for Intelligent Physical Agents
Fortran	Formula Translation
FTP	File Transport Protocol
GB	Gigabyte
GLPK	GNU Linear Programming Kit
GNU	GNU's Not Unix (rekursives Akronym)
GPS	Global Positioning System
GUID	Globally Unique Identifier
Hbf	Hauptbahnhof
HMD	Handbuch der maschinellen Datenverarbeitung
HTML	Hyper Text Markup Language
HTTP	Hyper Text Transport Protocol
IBM	International Business Machines Corporation
IBNR	Internationale Bahnstationsnummer
ICCL	International Committee on Computational Linguistics
ICCV	International Conference on Computer Vision
ID	Identifier
IDE	Integrated Development Environment
IEE	Institution of Electrical Engineers
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force

IP	Internet Protocol
IPv4	Internet Protocol Version 4
IPv6	Internet Protocol Version 6
IRC	Internet Relay Chat
ISO	International Organization for Standardization
IT	Informationstechnik
JIT	Just In Time
JSON	Javascript Object Notation
JsonML	JSON Markup Language
JSONx	JSON extended
JVM	Java Virtual Machine
KV	kombinierter Verkehr
Lkw	Lastkraftwagen
LP	Linear Programming
MAS	Multi-Agenten-System
MLJSON	MarkLogic JSON
MSIL	Microsoft Intermediate Language
MVC	Model View Controller
NHM	nomenclature harmonisée des marchandises
NP	nichtdeterministisch polynomiale Zeit
ODbL	Open Database Licence
OFB	Online-Frachtenbörse
ORFE	Online Rail Freight Exchange
OrGoLo	Organisatorische Innovationen mit Good Governance in Logistiknetzwerken
OS	Operating System
o.S.	ohne Seite
OSM	OpenStreetMap
PC	Personal Computer

PHP	PHP: Hypertext Preprocessor (rekursives Akronym)
RDBMS	Relational Database Management System
RFC	Request for Comments
S.	Seite
SCM	Software Configuration Management
SGML	Standard Generalized Markup Language
SMS	Short Message Service
sog.	sogenannte
SQL	Structured Query Language
SSL	Secure Sockets Layer
s.t.	subject to
SVN	Subversion
TCP	Transmission Control Protocol
u.	und
u.a.	unter anderem
UIC	union internationale des chemins de fer
UID	Unique Identifier
UMTS	Universal Mobile Telecommunications System
UN	United Nations
URL	Uniform Resource Locator
URN	Uniform Resource Name
UTC	Universal Time Coordinated
UUID	Universal Unique Identifier
VBScript	Visual Basic Script
VDV	Verband Deutscher Verkehrsunternehmen
vgl.	vergleiche
W3C	World Wide Web Consortium
WI	Wirtschaftsinformatik

WW	Warenwirtschaft
XHTML	Extensible Hypertext Markup Language
XML	Extensible Markup Language
XSLTJSON	Extensible Stylesheet Language Transformation JSON
z.B.	zum Beispiel

Symbolverzeichnis

\forall	für alle
\in	Element
α	Binärvariable für Gebotsausschluss
λ	zugeordnete Quantität eines Gutes
\sum	Summe
B	Gebot
G	Gesamttransport
H	Haltepunkt
M	Menge der gehandelten Güter
T	Termin
U	Menge der verfügbaren Quantitäten
Y	Menge der abgegebenen Gebote
a	Laufindex
b	Laufindex
e	Laufindex
g	Teildienstleistung in einem Gesamttransport
i	Laufindex
j	Bündel
k	Gut
m	Anzahl der Güter
n	Anzahl der Gebote
p	Preis
q	Gewicht
r	Radius bei flexiblen Transporten
s	Radius bei flexiblen Transporten
t	Anzahl der Teildienstleistungen in einem Gesamttransport
u	verfügbare Quantität eines Gutes
v	Volumen
x	Binärvariable für Gebotszuschlag

Abbildungsverzeichnis

1	Transport- und Ressourcenmarkt im Schienengüterverkehr	9
2	Benutzeroberfläche des Softwareprototyps aus dem EU-Projekt CODE24 . .	45
3	Übersicht über vorhandene Transportangebote	47
4	Eingabemaske für ein Transportangebot	48
5	Detailansicht eines Transportangebots	49
6	Zahlungsbereitschaft der potentiellen Nutzer einer Online-Frachtenbörse . .	53
7	Arten elektronischer Marktplätze aus Stakeholder-Perspektive	66
8	Klassifizierungen von Agenten	76
9	Sprachtypen zur Programmierung von Agenten	81
10	Umgebung in Multi-Agenten-Systemen	85
11	Konzeptionelle Struktur eines Multi-Agenten-Systems	86
12	Generisches Beispiel eines Ziel-Plan-Baums	89
13	Grafisches Beispiel für eine Standard-Auktion	98
14	Grafisches Beispiel für eine einseitige kombinatorische Auktion	101
15	Grafisches Beispiel für eine Börse	110
16	Beispiel für eine zentrale Architektur	117
17	Beispiel für eine dezentrale Architektur	118
18	Kontaktaufnahme unter Agenten mit Hilfe eines Funkfeuer-Agenten	120
19	Kontaktaufnahme unter Agenten mit Hilfe eines Rundruf-Kanals	121
20	Feste und flexible Haltepunkte	125
21	Flexible Haltepunkte	127
22	Haltepunkte – Spezialfall 1	128
23	Haltepunkte – Spezialfall 2	129
24	Haltepunkte – Spezialfall 3	130
25	Haltepunkte – Spezialfall 4	131
26	Exemplarisches Verkehrsnetz	132
27	Einfaches Beispiel zur Gruppenbildung	139
28	Erweitertes Beispiel zur Gruppenbildung	144
29	Beispiel für eine Ad-hoc-Auktion	147
30	Beispiel zur Vollständigkeit von Gruppen	148
31	Gruppenphasen im Rahmen von Ad-hoc-Auktionen	150
32	Startansicht der Benutzeroberfläche	192
33	Beispielhafte Ansicht zu Projekteigenschaften	194
34	Beispielhafte Kartenansicht im Karten-Modus	195
35	Beispielhafte Kartenansicht im Karten-Modus inklusive der Eigenschaften eines festen Haltepunkts	196
36	Webseite des OpenStreetMap-Projekts	198
37	Abstrakte Verbindungen und tatsächliche Streckenverläufe	200
38	Beispielhafte Kartenansicht im Agenten-Modus inklusive Agenteneigenschaf- ten	202
39	Beispielhafte Kartenansicht im Agenten-Modus inklusive der Eigenschaften eines festen Haltepunkts	203
40	Anlegen einer Angebotspräferenz für einen fixen Transport im Agenten- Modus	205
41	Anlegen einer Angebotspräferenz für einen flexiblen Transport im Agenten- Modus	206
42	Erstellung einer Nachfragepräferenz im Agenten-Modus	207
43	Übersicht über Szenarien	210
44	Beispielhafte Vorbereitung eines Szenarios zur Simulation	211

45	Startansicht einer Simulation	212
46	Zusammenfassung der Ergebnisse einer Simulation	213
47	Detailliertes Protokoll einer Simulation	214
48	Kartenansicht im Szenario „Spezialfall 1“	233
49	Kartenansicht im Szenario „Spezialfall 2“	234
50	Kartenansicht im Szenario „Spezialfall 3“	235
51	Kartenansicht im Szenario „Spezialfall 4“	237

Tabellenverzeichnis

1	Prognosen zur Güterbeförderungsleistung in Deutschland (in Mrd. Tonnenkilometern)	39
2	Preis- und Geschäftsmodelle ausgewählter Lkw-Online-Frachtenbörsen . . .	40
3	Zahlenbeispiel für eine Börse	110
4	Zahlenbeispiel für ein invalides Szenario	111
5	Beispiel für Gebote	136
6	Übersicht über die Gruppenphasen im Rahmen von Ad-hoc-Auktionen . .	149
7	Beurteilung von Agentenprogrammiersprachen bezüglich ihrer Eignung zur Implementierung des Softwareprototyps	183
8	Beurteilung von Datenformaten bezüglich ihrer Eignung zur Implementierung des Softwareprototyps	184
9	Beurteilung von Datenbanksystemen bezüglich ihrer Eignung zur Implementierung des Softwareprototyps	186
10	Beurteilung von Webframeworks bezüglich ihrer Eignung zur Implementierung des Softwareprototyps	187
11	Beurteilung von Webservern bezüglich ihrer Eignung zur Implementierung des Softwareprototyps	188
12	Beurteilung von Software zur Versionsverwaltung bezüglich ihrer Eignung zur Implementierung des Softwareprototyps	189
13	Erläuterung zur Agenten-Notation für Transportabschnitte	218
14	Erläuterung zur Agenten-Notation für Präferenzen	219
15	Erläuterung zum beispielhaften Transportabschnitt in Agenten-Notation .	220
16	Erläuterung zur beispielhaften Anbieterpräferenz in Agenten-Notation . .	221
17	Zahlenbeispiel für eine Börse mit ALT-Werten	226
18	Zahlenbeispiel für eine ALT-Wert-Matrix	227
19	Angebote und Nachfragen im Szenario „Spezialfall 1“	238
20	Angebote und Nachfragen im Szenario „Spezialfall 2“	239
21	Angebote und Nachfragen im Szenario „Spezialfall 3“	240
22	Angebote und Nachfragen im Szenario „Spezialfall 4“	241
23	Beschreibung der Datentypen in der Datenbank	274
24	Beschreibung der Datenbanktabelle „agent_configurations“	275
25	Beschreibung der Datenbanktabelle „agent_address_book_entries“	275
26	Beschreibung der Datenbanktabelle „agent_config_simulation_scenarios“ .	275
27	Beschreibung der Datenbanktabelle „agent_transport_offer_sections“ . . .	276
28	Beschreibung der Datenbanktabelle „agent_transport_offers“	276
29	Beschreibung der Datenbanktabelle „agent_transport_requests“	277
30	Beschreibung der Datenbanktabelle „delayed_jobs“	278
31	Beschreibung der Datenbanktabelle „projects“	278
32	Beschreibung der Datenbanktabelle „simulation_metrics“	279
33	Beschreibung der Datenbanktabelle „simulation_scenario_runs“	279
34	Beschreibung der Datenbanktabelle „simulation_scenarios“	279
35	Beschreibung der Datenbanktabelle „transport_network_relations“	280
36	Beschreibung der Datenbanktabelle „transport_network_stations“	280
37	Beschreibung der Datenbanktabelle „transport_network_way_points“ . . .	281

Listingverzeichnis

1	Beispielhafte Nachricht in XML-Notation	167
2	Beispielhafte Nachricht in JSON-Notation	169
3	Beispielhafte Nachricht in DSL-Notation	171
4	Agenten-Notation für Transportabschnitte	218
5	Agenten-Notation für Präferenzen	219
6	Beispielhafter Transportabschnitt in Agenten-Notation (1)	220
7	Beispielhafter Transportabschnitt in Agenten-Notation (2)	221
8	Beispielhafte Anbieterpräferenz in Agenten-Notation	221
9	Beispielhafte Anbieterpräferenz mit mehreren Transportabschnitten in Agenten-Notation	222
10	Beispielhafte Anbieterpräferenz mit alternativen Angeboten in Agenten-Notation	222

1 Einleitung

1.1 Forschungsmotivation und Zielsetzung

Der europäische Schienengüterverkehr befindet sich im Wandel. Angestoßen durch den von der Europäischen Union im Jahr 1994 begonnenen Deregulierungsprozess findet seit zwei Jahrzehnten ein Wettbewerb statt, der wenige Jahre zuvor ebenso unwahrscheinlich schien wie die durch ihn entstandenen Nischenmärkte.¹

Im Rahmen der Bemühungen der Europäischen Union (EU), den Güterverkehr aus verkehrs- und umweltpolitischen Gründen von der Straße auf die Schiene zu verlagern, werden Infrastrukturprojekte wie das „Projekt CODE24“ gefördert.² Im Rahmen dieses Projektes war unter anderem die Schaffung einer Online-Frachtenbörse³ zum elektronischen Handel von güterspezifischen Transportdienstleistungen vorgesehen.⁴

Das Teilprojekt „Entwicklung einer Online-Güterbörse für Transportleistungen im Schienengüterverkehr“ konnte Fortschritte u.a. bei der prototypischen Implementierung eines internetbasierten Softwaresystems vorweisen.⁵ Jedoch wurden im Laufe der Realisierung dieser Plattform strukturelle Nachteile der Etablierung einer Online-Frachtenbörse für Transportleistungen im Schienengüterverkehr deutlich. Einige dieser Nachteile sind direkt dem Verkehrsträger Schiene und den tradierten, teils hegemoniellen Strukturen der Schienengüterverkehrsbranche zuzurechnen, wie bspw. die ausgeprägten Informationsasymmetrien zwischen Anbietern und Nachfragern von Transportdienstleistungen sowie die Notwendigkeit des Vor- und Nachlaufs auf der Straße.⁶

Die Anforderungen an eine Online-Frachtenbörse für den Schienengüterverkehr orientieren sich stark an den Anforderungen des Verkehrsträgers Schiene und der mit ihm verbundenen Branchenstrukturen.⁷ Zusätzlich beinhalten sie auch regelmäßig die Forderung nach

¹ Vgl. FISCHER (2008), S. 1 f.

² Vgl. SCHECK/WILSKE (2011), S. 64 f.

³ In verschiedenen Publikationen finden sich die Begriffe „Online-Frachtenbörse“, „Online-Transportbörse“ sowie „Online-Güterbörse“ zur Beschreibung einer internetbasierten Börse zur Vermittlung von Angeboten von und Nachfragen nach Transportdienstleistungen für Sachgüter. In dieser Forschungsarbeit wird der Begriff „Online-Frachtenbörse“ synonym für alle drei Begriffe genutzt.

⁴ Vgl. SCHECK/WILSKE (2011), S. 67.

⁵ Im Rahmen des Projekts CODE24 wurden am Institut für Produktion und Industrielles Informationsmanagement der Universität Duisburg-Essen mehrere Forschungsarbeiten veröffentlicht, welche sich mit der Etablierung einer Online-Frachtenbörse für den europäischen Schienengüterverkehr beschäftigen. Vgl. BRUNS et al. (2012a), BRUNS et al. (2012b), HABIB/BRUNS (2012), FÖHRING/BRUNS (2011), KLIPPERT et al. (2011), WEICHELT/BRUNS (2011), sowie BRUNS et al. (2010a).

⁶ Eine detaillierte Analyse der relevanten Realprobleme im Schienengüterverkehr erfolgt in Kapitel 2.2. Vgl. Kapitel 2.2, S. 9.

⁷ Eine detaillierte Darstellung der Anforderungen an eine Online-Frachtenbörse für den Schienengüterverkehr erfolgt in Kapitel 2.4.2. Vgl. Kapitel 2.4.2, S. 17.

einer Berücksichtigung des Vor- und Nachlaufs der schienengebundenen Transporte über andere Verkehrsträger (am häufigsten per Lkw).⁸

Hierdurch wird offenkundig, dass eine Erfolg versprechende Online-Frachtenbörse für den Schienengüterverkehr eine multimodale Frachtenbörse sein müsste, welche in der Lage ist, mehrere Verkehrsträgerarten bei der Durchführung kombinierter Verkehre⁹, zu berücksichtigen. Dies legt den Schluss nahe, dass ein Konzept für eine über schienengebundene Transporte hinausgehende Online-Frachtenbörse betriebswirtschaftlich wünschenswert ist.¹⁰

Die Forschungsmotivation dieser Forschungsarbeit besteht darin – ausgehend von der bereits im EU-Projekt CODE24 erfolgten Forschung im Bereich „Online-Frachtenbörse für Transportleistungen im Schienengüterverkehr“ und unter Berücksichtigung der bereits gewonnenen Erkenntnisse – einen virtuellen Marktplatz zu konzipieren, welcher den Handel multimodaler Gütertransporte ermöglicht, während er möglichst viele der beschriebenen Nachteile aufhebt und die Einstiegshürden für potentielle Marktplatzteilnehmer so weit wie konzeptionell möglich senkt.

Die Zielsetzung dieser Forschungsarbeit ist die prototypische Implementierung dieses Marktplatzes in Form eines internetbasierten Softwaresystems. Es soll eruiert werden, wie sich die erstellten Konzepte in die Praxis übersetzen lassen und welche Probleme und Weiterentwicklungspotentiale bei ihrer Implementierung identifiziert werden können. Der im Rahmen dieser Forschungsarbeit implementierte Softwareprototyp soll die technische Machbarkeit des konzipierten Marktplatzes demonstrieren und weitere Forschung in Richtung einer dezentral organisierten Frachtenbörse für den kombinierten Verkehr ermöglichen.¹¹

⁸ Vgl. HABIB/BRUNS (2012), S. 26 u. 29, sowie ENDEMANN/KASPERS (2011), S. 32, 33 u. 35.

⁹ Sogenannte „kombinierte Verkehre“ bezeichnen eine Unterart multimodaler Verkehre. Der Begriff „multimodaler Verkehr“ bezeichnet einen Transport derselben Güter mit mehreren verschiedenartigen Verkehrsträgern. Der Begriff „intermodaler Verkehr“ bezeichnet einen multimodalen Verkehr, wobei die Güter in derselben Ladeinheit oder demselben Straßenfahrzeug transportiert werden und ein Wechsel des Verkehrsträgers, aber kein Umschlag der transportierten Güter selbst erfolgt. Vgl. UNITED NATIONS (2001), S. 16 f. Wenn der hauptsächliche Teil eines intermodalen Verkehrs mit der Eisenbahn, dem Binnen- oder Seeschiff bewältigt und der Vor- und Nachlauf auf der Straße so kurz wie möglich gehalten wird, so bezeichnet man dies als „kombinierten Verkehr“. Vgl. UNITED NATIONS (2001), S. 18.

¹⁰ Eine detaillierte Darstellung der Desiderata erfolgt in Kapitel 3. Vgl. Kapitel 3, S. 55.

¹¹ Eine genaue Auflistung der wissenschaftlichen Probleme und intendierten Ergebnisse erfolgt in Kapitel 4. Vgl. Kapitel 4, S. 59.

1.2 Forschungsstand

In der Fachliteratur wird dem Thema „Online-Frachtenbörse für den Schienengüterverkehr“ erst in jüngster Vergangenheit größere Beachtung geschenkt.¹²

In der betrieblichen Praxis existierte bei Beginn der Forschungsarbeiten im Jahr 2011 im europäischen Raum – im Gegensatz zum straßengebundenen Güterverkehr – keine dedizierte Frachtenbörse für den Schienengüterverkehr.¹³ Zudem mangelte es – unabhängig vom betrachteten Verkehrsträger – an einer europaweit ausgerichteten Online-Frachtenbörse. Auch auf nationaler Ebene ließen sich nur wenige Frachtenbörsen identifizieren, welche als Konkurrenz zu einer neu am Markt auftretenden europaweiten Online-Frachtenbörse anzusehen gewesen wären.¹⁴

Seit dem Jahr 2010 werden vermehrt Anstrengungen unternommen, die Anforderungen an eine Online-Frachtenbörse für den Schienengüterverkehr zu untersuchen.

KLIPPERT et al. haben eine Anforderungsanalyse für eine Online-Frachtenbörse im Schienengüterverkehr vorgenommen.¹⁵ Es konnten Erfolgsfaktoren und Anforderungen identifiziert und ein Leistungskatalog erstellt werden, auf deren Basis ein erster Softwareprototyp entwickelt wurde.¹⁶ Dieser in den Jahren 2010 bis 2012 entwickelte Softwareprototyp einer Online-Frachtenbörse für den Schienengüterverkehr wird als „ORFE-Softwareprototyp“ („Online Rail Freight Exchange“) bezeichnet.¹⁷

HABIB und BRUNS haben das kompetitive Umfeld einer möglichen Online-Frachtenbörse analysiert.¹⁸ Es wurde eine Stakeholderanalyse durchgeführt, welche die Einschätzung ausgewählter Domänenexperten zu den Erfolgsaussichten und Anforderungen einer Online-Frachtenbörse untersuchte.¹⁹

¹² Vgl. KLIPPERT et al. (2013), FIDAN/KUHLMANN (2013), BRUNS et al. (2012a), BRUNS et al. (2012b), FÖHRING et al. (2012), HABIB/BRUNS (2012), ENDEMANN/KASPERS (2011), FÖHRING/BRUNS (2011), KLIPPERT et al. (2011), WEICHEL/BRUNS (2011), sowie BRUNS et al. (2010a).

¹³ Vgl. ENDEMANN/KASPERS (2011), S. 32.

¹⁴ Die als gering eingeschätzte Konkurrenzfähigkeit der bestehenden Online-Frachtenbörsen folgt aus der Plausibilitätsüberlegung, dass diese nur national agieren, wodurch am Markt eine Nische für eine europaweit ausgerichtete Online-Frachtenbörse entsteht.

¹⁵ Vgl. KLIPPERT et al. (2011).

¹⁶ Eine zusammenfassende Darstellung der Entwicklung des ORFE-Softwareprototyps im Rahmen des EU-Projekts CODE24 wird in Kapitel 2.4.6 gegeben. Vgl. Kapitel 2.4.6, S. 44. Ausführlichere Darstellungen der Konzipierung, Entwicklung und Rezeption wurden ebenfalls veröffentlicht. Vgl. FÖHRING/ZELEWSKI (2013), FÖHRING et al. (2012), FÖHRING/BRUNS (2011), sowie KLIPPERT et al. (2011).

¹⁷ Eine kritische Reflexion der Entwicklung des ORFE-Softwareprototyps im Rahmen des EU-Projekts CODE24 wird in Kapitel 2.4.7 gegeben. Vgl. Kapitel 2.4.7, S. 50.

¹⁸ Vgl. HABIB/BRUNS (2012).

¹⁹ Ein Überblick über die Stakeholderanalyse wird in Kapitel 2.4.3 gegeben. Vgl. Kapitel 2.4.3, S. 30.

BRUNS et al. haben das Marktpotential einer derartigen Online-Frachtenbörse näher untersucht.²⁰ Es wurde das Potential einer Online-Frachtenbörse für den Schienengüterverkehr vor dem Hintergrund der prognostizierten Entwicklung des Güterverkehrsaufkommens in Europa vorgenommen. Zusätzlich wurden Experten zu ihrer Einschätzung verschiedener Szenarien befragt.²¹

Während der Anfertigung dieser Forschungsarbeit wurden zwei Online-Frachtenbörsen gestartet. Beiden Unternehmen wurde der „ORFE-Softwareprototyp“ samt aller weiteren CODE24-Ergebnisse als Arbeitsgrundlage zur Verfügung gestellt.²² Es handelt sich um die Plattformen „Railcargo-Online“²³ sowie „Freit-One“^{24, 25}.

Zudem haben sich in der betrieblichen Praxis erste Online-Frachtenbörsen etabliert, deren Zielsetzung bzgl. der Vermittlung von Angeboten von und Nachfragen nach multimodalen Gütertransportdienstleistungen der Zielsetzung dieser Forschungsarbeit ähnlich ist.²⁶ Die Produkte dieser Unternehmen haben jedoch eine von den Zielen dieser Arbeit abweichende Fokussierung, welche auf einer besseren Integration der insbesondere im Straßengüterverkehr bereits vorhandenen Telemetrie, der Einbindung von mobilen Endgeräten wie Mobiltelefonen sowie der Vermittlung von komplementären Dienstleistungen liegt.²⁷

1.3 Aufbau der Forschungsarbeit

In Kapitel 2 werden die wissenschaftlichen und wirtschaftlichen Rahmenbedingungen, innerhalb derer die Forschungsarbeit entsteht, vorgestellt. Es werden insbesondere die Bemühungen der Europäischen Union beschrieben, im Rahmen des Projekts CODE24 eine Online-Frachtenbörse für den europäischen Schienengüterverkehr zu etablieren. Es wird aufgezeigt, aus welchen Gründen die von den befragten Experten gewünschte Online-Frachtenbörse konzeptionell eine multimodale, über die Vermittlung von Schienengüterverkehren und kombinierten Verkehren hinausgehende, Frachtenbörse sein muss.

In Kapitel 3 werden aus den aufgezeigten Widersprüchen und Missständen wünschenswerte Eigenschaften einer idealtypischen Online-Frachtenbörse abgeleitet und in Form betriebswirtschaftlicher Desiderata besprochen.

²⁰ Vgl. BRUNS et al. (2012a).

²¹ Ein Überblick über die Untersuchung des Marktpotentials wird in Kapitel 2.4.5 gegeben. Vgl. Kapitel 2.4.5, S. 38.

²² Vgl. DÖRR/ENDEMANN (2014), S. 61 f.

²³ Weitere Informationen sind im Internet verfügbar. Vgl. RAILCARGOONLINE (2017).

²⁴ Weitere Informationen sind im Internet verfügbar. Vgl. FREITONE (2017).

²⁵ Vgl. DÖRR/ENDEMANN (2014), S. 61 f.

²⁶ Vgl. BOTTLER (2015), S. 16 f.

²⁷ Vgl. BOTTLER (2015), S. 16 f.

In Kapitel 4 werden aus den betriebswirtschaftlichen Desiderata die konkreten wissenschaftlichen Probleme dieser Forschungsarbeit definiert.

In Kapitel 5 werden die Grundlagen zu elektronischen Marktplätzen, Multi-Agenten-Systemen und Auktionen erläutert. Es wird gezeigt, welche Relevanz diese Teilgebiete für die Lösung der wissenschaftlichen Probleme besitzen.

In Kapitel 6 werden die, zur Entwicklung der autonomen Software-Agenten entwickelten, Konzepte verteilter Marktplätze, Ad-hoc-Auktionen sowie des abstrakten Verkehrsnetzes vorgestellt.

Kapitel 7 beschreibt, welche Werkzeuge im Rahmen dieser Forschungsarbeit implementiert wurden, um die prototypische Implementierung einer Online-Frachtenbörse zu ermöglichen. Es wird aufgezeigt, inwieweit bestehende Werkzeuge mit speziell entwickelten Werkzeugen kombiniert wurden, um eine Entwicklungsumgebung zu schaffen, welche die anschließend beschriebene Implementierung von autonomen Software-Agenten unterstützt haben.

In Kapitel 8 werden die Installation und der Betrieb der implementierten Software beschrieben, um nachfolgenden Forschungsarbeiten die Arbeit mit derselben zu erleichtern und Einstiegshürden bei der Erprobung des Softwareprototyps zu senken.

In Kapitel 9 wird anhand von ausgewählten Beispielen demonstriert, wie mit Hilfe des Softwareprototyps Simulationen von kombinierten Verkehren durchgeführt werden können.

In Kapitel 10 werden die wesentlichen Ergebnisse dieser Forschungsarbeit zusammengefasst und vor dem Hintergrund weitergehender Forschung kritisch reflektiert sowie Weiterentwicklungspotentiale der vorgestellten Konzepte und Werkzeuge präsentiert.

2 Rahmenbedingungen für die Etablierung einer europäischen Online-Frachtenbörse

2.1 Hintergrund am Beispiel der Entwicklung des Schienengüterverkehrs

Im vorliegenden Kapitel werden die Hintergründe jener Realprobleme dargelegt, welche diese Forschungsarbeit motivieren. Um diese Realprobleme und die aus ihnen resultierenden wissenschaftlichen Probleme besser darlegen zu können, werden zunächst ausgewählte historische Entwicklungen und Forschungsergebnisse zum Transportwesen in Europa am Beispiel des Schienengüterverkehrs erläutert.

Der europäische Schienengüterverkehr befindet sich seit Jahren im Wandel. Der von der Europäischen Union verordnete Deregulierungsprozess zur rechnerischen und organisatorischen Trennung von Fahrweg und Eisenbahnbetrieb sowie zur unabhängigen Geschäftsführung der Eisenbahnen ohne staatliche Kontrolle hat das Wettbewerbsumfeld nachhaltig verändert.²⁸

Während heute der Markt die Bedingungen von schienengebundenen Güterverkehren bestimmt, wurden bis zur Liberalisierung im Jahr 1994 Transportkonditionen und -preise von staatlicher Seite diktiert.²⁹ Die Möglichkeiten zur Nutzung fremder Infrastruktur und zur Kooperation verschiedener Eisenbahnen waren zuvor nicht gegeben.³⁰ Erst die Liberalisierung des Marktes für Eisenbahngüterverkehre ermöglichte es, dass Eisenbahnen ohne eigene Infrastrukturen betrieben werden konnten.³¹ Gleichzeitig wurden die staatlichen Bahnen zu einer anhand von betriebswirtschaftlichen Grundsätzen ausgerichteten Unternehmensführung gezwungen.³² FISCHER argumentiert, dass ein seit 1970 zunehmender Bedeutungsverlust des Verkehrsträgers Schiene den Anreiz zur Deregulierung gab.³³

Insbesondere bei der Fokussierung auf „Just In Time“-Konzepte konnten sich Transporte über die Schiene immer schwerer gegen Transporte per Lkw behaupten.³⁴ Doch während der Verkehrsträger Schiene durch Veränderungen in den Wertschöpfungsketten der produzierenden Unternehmen für diese an Attraktivität verlor, entstand infol-

²⁸ Vgl. FISCHER (2008), S. 1.

²⁹ Vgl. WIELAND (2010), S. 47 ff.

³⁰ Vgl. HABIB/BRUNS (2012), S. 7.

³¹ Vgl. HABIB/BRUNS (2012), S. 7.

³² Vgl. FISCHER (2008), S. 1.

³³ Vgl. FISCHER (2008), S. 15.

³⁴ Vgl. KÖNIG/JUGELT (2009), S. 115.

ge des Deregulierungsprozesses ein Markt für Bahnleistungen, der zuvor nicht existierte.³⁵

Heute leidet der europäische Schienengüterverkehr zunehmend daran, dass die produzierenden Unternehmen vermehrt Transportanforderungen stellen, welche durch die bestehende Infrastrukturen im Schienengüterverkehr nur begrenzt erfüllt werden können.³⁶ Der Schienengüterverkehr ist darauf angewiesen, dass bereits bei der Planung von Produktionsstätten und Umschlagterminals eine Anbindung an das Schienennetz und die Bereitstellung angepasster Transportmittel – bspw. spezieller Wagen – berücksichtigt werden.³⁷

Transporte werden im schienengebundenen Güterverkehr von Eisenbahnverkehrsunternehmen (EVU) durchgeführt.³⁸ Diese Unternehmen erhalten Aufträge zum Transport von Sachgütern entweder direkt von den Industrie- und Handelsunternehmen oder von mit dem Transport beauftragten Speditionen.³⁹ Die transportierten Sachgüter umfassen Rohstoffe, wie bspw. Kohle und Stahl, Konsumgüter, wie bspw. chemische Erzeugnisse oder Lebensmittel, sowie Abfälle.⁴⁰ Die Transporte werden unter Verwendung von Wagen, Transportbehältern und verschiedenen Umschlagtechniken durchgeführt.⁴¹

Vor dem Hintergrund steigenden Kostendrucks auf alle Bereiche der Wertschöpfungskette und die Eisenbahnverkehrsunternehmen rückt die effizientere Nutzung bestehender Infrastrukturen und weiterer Ressourcen mehr und mehr in den Fokus der Unternehmen und Entscheider.⁴² Die intensivere Nutzung von Gleisinfrastruktur, Betriebsmitteln, Personal, Energie und auch Informationen spielt hierbei eine entscheidende Rolle.⁴³ Die Herausforderungen, denen sich die Eisenbahnverkehrsunternehmen stellen müssen, sind nicht nur ein wettbewerbsfähiger Preis und eine zuverlässige Transportdurchführung, sondern auch der zeitgenaue Versand sowie ein sequenzgenauer Eingang der Güter in passfähigen Transportbehältern inklusive aller relevanten Status- und Transportgüterinformationen.⁴⁴ Insbesondere die Verfügbarkeit von passfähigen Transportbehältern zur nahtlosen Einbindung der Güterverkehre in hochautomatisierte Produktionsprozesse kann dem Schie-

³⁵ Vgl. FISCHER (2008), S. 2.

³⁶ Vgl. FISCHER (2008), S. 1.

³⁷ Vgl. KÖNIG/JUGELT (2009), S. 115 u. 117.

³⁸ Vgl. BUTTERMANN (2002), S. 70.

³⁹ Vgl. BUTTERMANN (2002), S. 64 ff.

⁴⁰ Vgl. BUTTERMANN (2002), S. 64 ff.

⁴¹ Vgl. BUTTERMANN (2002), S. 64 ff.

⁴² Vgl. KÖNIG/JUGELT (2009), S. 116.

⁴³ Vgl. KÖNIG/JUGELT (2009), S. 116.

⁴⁴ Vgl. KÖNIG/JUGELT (2009), S. 115.

nengüterverkehr im Rahmen von „Just-in-Sequence“-Konzepten einen Vorteil gegenüber straßengebundenen Transporten ermöglichen.⁴⁵

Sollten der Ausgangs- oder Bestimmungsort nicht über einen direkten Gleisanschluss verfügen, so müssen die transportierten Güter an bestehenden Terminals umgeschlagen werden.⁴⁶ Oftmals wird der Vor- und Nachlauf zur Schiene über die Straße vollzogen.⁴⁷

Es ist auch im Schienengüterverkehr, analog zum Straßengüterverkehr, üblich bei der Organisation von Transporten eine Spedition zwischenzuschalten, wenngleich die traditionelle Art der Geschäftsanbahnung im Schienengüterverkehr die Kommunikation zwischen Verloader und Frachtführer ist.⁴⁸ Speditionen nehmen im Transportwesen sowohl die Rolle des Anbieters als auch des Nachfragers ein.⁴⁹ Sie befinden sich in der Anbieterrolle, wenn produzierende Unternehmen sich nicht direkt an ein Eisenbahnverkehrsunternehmen wenden, sondern eine Spedition mit der Vermittlung beauftragen.⁵⁰ Sie befinden sich in der Nachfragerrolle, wenn sie Ressourcen bei Eisenbahnverkehrsunternehmen einkaufen.⁵¹ Nehmen Güter verschiedener Spediteure an einem Transport per Zug teil, so werden häufig Operateure eingeschaltet, welche als Hauptspediteure für den gesamten Zug auftreten.⁵²

Somit ergeben sich die verschiedenen Kombinationen, unter denen ein Eisenbahntransport zwischen produzierenden Unternehmen, Spediteuren und Operateuren sowie Eisenbahnverkehrsunternehmen zustande kommen kann. Die genannten Akteure bilden den sog. „Transportmarkt“ und fragen die für die zu organisierenden Transporte relevanten Ressourcen auf dem sog. „Ressourcenmarkt“ nach.⁵³

Abbildung 1 stellt die Akteure des Transportmarktes und ihre wirtschaftlichen Beziehungen untereinander sowie ihren regelmäßigen Zugriff auf Komponenten des Ressourcenmarktes schematisch dar.

Der Ressourcenmarkt umfasst all jene Komponenten, welche die Eisenbahnverkehrsunternehmen zur Durchführung ihrer Transporte benötigen.⁵⁴ Hierzu gehört die Transportkette,

⁴⁵ Vgl. KÖNIG/JUGELT (2009), S. 115.

⁴⁶ Vgl. BUTTERMANN (2002), S. 71 ff.

⁴⁷ Vgl. BUTTERMANN (2002), S. 71 ff.

⁴⁸ Vgl. HABIB/BRUNS (2012), S. 8.

⁴⁹ Vgl. HABIB/BRUNS (2012), S. 8.

⁵⁰ Vgl. HABIB/BRUNS (2012), S. 8.

⁵¹ Vgl. HABIB/BRUNS (2012), S. 8.

⁵² Vgl. HABIB/BRUNS (2012), S. 8.

⁵³ Vgl. HABIB/BRUNS (2012), S. 8 f.

⁵⁴ Vgl. HABIB/BRUNS (2012), S. 9.

welche sich durch die Kooperation von Eisenbahnverkehrsunternehmen untereinander bildet.⁵⁵ Da ein Gütertransport selten von einem einzigen Eisenbahnverkehrsunternehmen durchgeführt wird, sind Kooperationspartner insbesondere im kombinierten und internationalen Verkehr erforderlich.⁵⁶ Der Ressourcenmarkt dient weiterhin der Beschaffung der benötigten Fahrzeuge – wie bspw. Lokomotiven und Wagen – sowie des erforderlichen Personals, Bahnhöfe, Tankstellen, Werkstätten sowie im Falle multimodaler Verkehre von Terminals zum Umschlag der Güter.⁵⁷

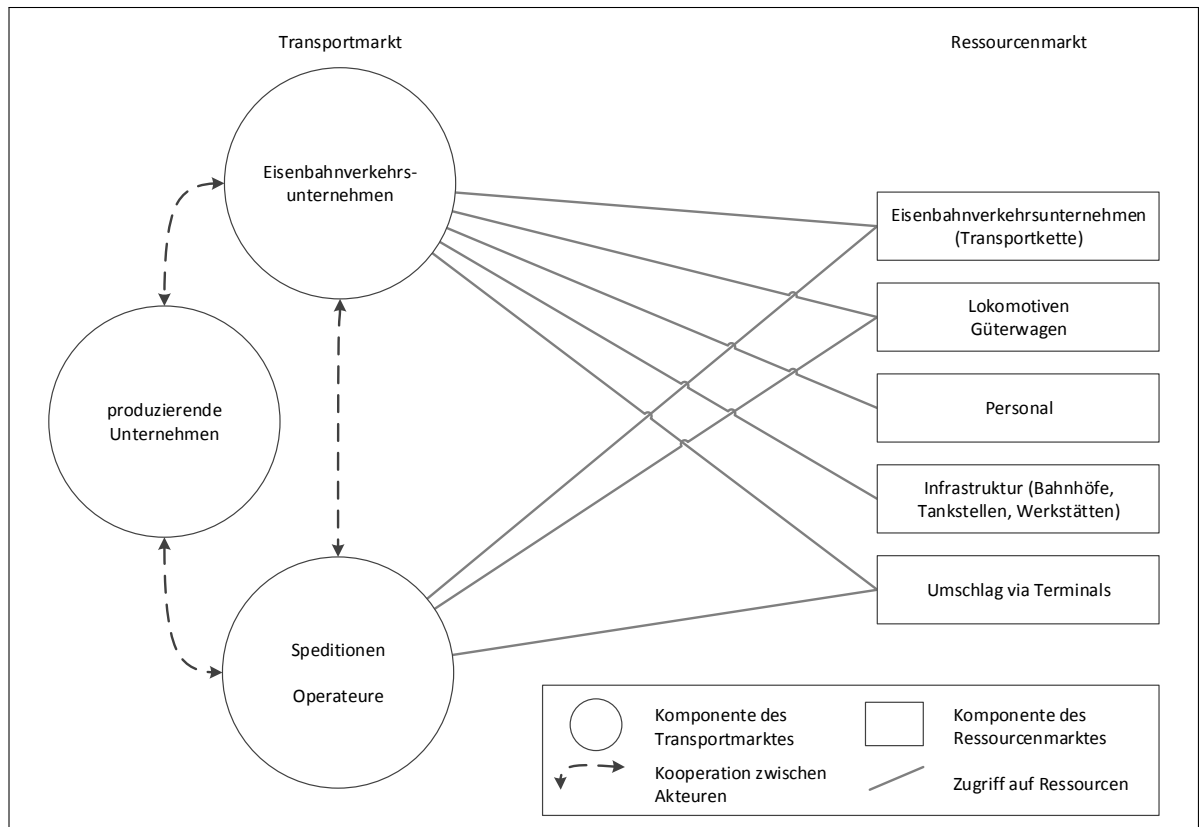


Abbildung 1: Transport- und Ressourcenmarkt im Schienengüterverkehr⁵⁸

Während Eisenbahnunternehmen in der Regel Zugriff auf alle genannten Ressourcen haben, gibt es auch Speditionen, welche benötigte Teilleistungen selbst einkaufen.⁵⁹

2.2 Realprobleme im Schienengüterverkehr

Die Situation im europäischen Güterverkehr aus Sicht der Schienengüterverkehrsakteure und die mit ihr einhergehenden Herausforderungen und Realprobleme stellen sich folgendermaßen dar:

⁵⁵ Vgl. HABIB/BRUNS (2012), S. 9.

⁵⁶ Vgl. HABIB/BRUNS (2012), S. 9.

⁵⁷ Vgl. HABIB/BRUNS (2012), S. 9.

⁵⁸ Quelle: eigene Darstellung in Anlehnung an HABIB/BRUNS (2012), S. 9.

⁵⁹ Vgl. HABIB/BRUNS (2012), S. 9.

Während im europäischen Schienengüterverkehr grundsätzlich Marktpotentiale vorhanden sind,⁶⁰ ist das Vorhandensein von Kapazitätsreserven unklar.⁶¹ Ein Kommunikationsmedium zwischen den beteiligten Akteuren des Schienengüterverkehrs fehlt.⁶²

Obwohl der Einzelwagenverkehr rund 50 Prozent des Schienengüterverkehrs in Europa umfasst, sind die Investitionen in die Schieneninfrastruktur rückläufig und es besteht kein europaweit einheitlicher Zugang zum bestehenden Schienennetz.⁶³ So besteht ein teils erschwelter Zugang zu Terminals und Tankstellen und nationale Betriebsvorschriften sowie gesetzliche Bestimmungen weichen ebenso voneinander ab wie die eingesetzte Betriebsprache.⁶⁴ Auch wurden Einzelwagenverkehre in bestimmten europäischen Ländern, bspw. Italien, reduziert oder komplett eingestellt.⁶⁵

Zudem mangelt es an der Gestaltung eines Geschäftsprozesses für die Schiene:⁶⁶ Die gesamte Informations- und Transportkette mit allen transport- und produktrelevanten Daten bedarf einer datenverarbeitungstechnischen Gestaltung, bei der Statusinformationen vom Kundenauftrag über die Produktübergabe bis zur Auslieferung beim Empfänger sowie die Frachtab- und -verrechnung lückenlos bereitgestellt werden.⁶⁷

Aus Kundensicht treten weitere Probleme auf:

Der Zugang zu Informationen der Eisenbahnverkehrsunternehmen ist aufwändig.⁶⁸ Es bestehen bei Frachtanfragen häufig lange Bearbeitungszeiten durch die Eisenbahnverkehrsunternehmen und die Preisangaben erfolgen regelmäßig auf Tonnenbasis anstelle von Angaben pro Transporteinheit.⁶⁹ Insbesondere bei grenzüberschreitenden Verkehren fallen gegenüber dem Straßengüterverkehr eine geringere Preistransparenz durch eine komplexere Preisbildung sowie Qualitätsmängel bei den Bedienungsstandards auf.⁷⁰ Der Schienengüterverkehr hat Probleme mit der Bereitstellung wettbewerbsfähiger Frachtpreise.⁷¹ Bei Lkw-Transporten lassen sich hingegen Preisschwankungen mit einer Tendenz zu Dumpingangeboten beobachten.⁷²

⁶⁰ Vgl. ENDEMANN/KASPERS (2011), S. 32.

⁶¹ Vgl. Kapitel 2.3, S. 13.

⁶² Vgl. ENDEMANN/KASPERS (2011), S. 32.

⁶³ Vgl. ENDEMANN/KASPERS (2011), S. 34.

⁶⁴ Vgl. ENDEMANN/KASPERS (2011), S. 34.

⁶⁵ Vgl. ENDEMANN/KASPERS (2011), S. 34.

⁶⁶ Vgl. ENDEMANN/KASPERS (2011), S. 35.

⁶⁷ Vgl. HABIB/BRUNS (2012), S. 36, sowie ENDEMANN/KASPERS (2011), S. 35.

⁶⁸ Vgl. ENDEMANN/KASPERS (2011), S. 34.

⁶⁹ Vgl. ENDEMANN/KASPERS (2011), S. 34.

⁷⁰ Vgl. ENDEMANN/KASPERS (2011), S. 34.

⁷¹ Vgl. ENDEMANN/KASPERS (2011), S. 34.

⁷² Vgl. ENDEMANN/KASPERS (2011), S. 34.

Die Folge ist, dass sich viele Kunden vom Verkehrsträger Schiene abwenden.⁷³

Darüber hinaus bestehen bei den Kunden weitere, individuelle Gründe für diesen Trend:⁷⁴ Auf Seiten der Anbieter von Transporten fehlen Ansprechpartner, bspw. um die Wagenstellung zu klären, und regelmäßig sorgt der fehlende Gleisanschluss für eine Entscheidung gegen den Transport per Schiene.⁷⁵

Es ist festzustellen, dass die produzierenden Unternehmen als Kunden – analog zum Straßengüterverkehr – eine transparente Kostenstruktur zur Vergleichbarkeit von Angeboten mit anderen Verkehrsträgern erwarten.⁷⁶ Ebenfalls analog zum Straßengüterverkehr erwarten die Verloader einen alleinverantwortlichen Dienstleistungspartner, der unabhängig von nationalen Grenzen und Zuständigkeiten für die gesamte Transportkette agiert.⁷⁷ Anfragen zu Frachtpreisen, Transportstatus sowie Frachtabrechnung sollten schnell beantwortet werden, denn die persönliche Betreuung spielt – neben der Vorhaltung der entsprechenden Informationen sowie der Einhaltung der vereinbarten Beförderungszeiten – für die Kunden eine entscheidende Rolle.⁷⁸

Doch ausgerechnet hier treffen die produzierenden Unternehmen im Straßengüterverkehr besser funktionierende Prozesse an.⁷⁹ Die persönliche Betreuung ist direkter, Angebote zu Frachtanfragen liegen schneller vor und auch Antworten auf Anfragen bzgl. des Status von Transportaufträgen fallen zufriedenstellender aus als bei vergleichbaren Anfragen an Eisenbahnverkehrsunternehmen.⁸⁰ Während sich auf nationaler Ebene eine Verbesserung der Situation andeutet, wünschen sich viele Kunden auch im internationalen Bahnverkehr – unabhängig von der Anzahl der beteiligten Staatsbahnen – schnell verfügbare, standardisierte Informationen über ihre Transportaufträge.⁸¹

Der Wunsch nach einer „Datendrehscheibe“ im Internet – einer Online-Frachtenbörse – wird geäußert.⁸²

⁷³ Vgl. ENDEMANN/KASPERS (2011), S. 34.

⁷⁴ Vgl. ENDEMANN/KASPERS (2011), S. 34.

⁷⁵ Vgl. ENDEMANN/KASPERS (2011), S. 34.

⁷⁶ Vgl. HABIB/BRUNS (2012), S. 40, sowie ENDEMANN/KASPERS (2011), S. 35.

⁷⁷ Vgl. ENDEMANN/KASPERS (2011), S. 35.

⁷⁸ Vgl. ENDEMANN/KASPERS (2011), S. 35.

⁷⁹ Vgl. ENDEMANN/KASPERS (2011), S. 35.

⁸⁰ Vgl. ENDEMANN/KASPERS (2011), S. 35.

⁸¹ Vgl. ENDEMANN/KASPERS (2011), S. 35.

⁸² Vgl. ENDEMANN/KASPERS (2011), S. 35.

2.3 Das Projekt CODE24

Vor dem Hintergrund anhaltender Globalisierungstendenzen erscheint ein Anstieg der Frachtraten an den europäischen Seehäfen als wahrscheinlich.⁸³ Bei Eintritt dieser Prognose ist analog von einem Anstieg der Frachtenraten in den Hinterländern dieser Häfen auszugehen. Aufgrund dieser Überlegungen sowie der im vorangegangenen Kapitel beschriebenen Ausgangslage startete im Rahmen des INTERREG-IVB-Programms der Europäischen Union zu Beginn des Jahres 2010 das „Projekt CODE24“.⁸⁴

Das Akronym „CODE24“ steht für „Corridor Development 24“ und nimmt Bezug auf die Region des Korridors 24, welcher die Häfen von Rotterdam und Antwerpen an der Nordsee mit dem Hafen Genua am Mittelmeer verbindet.⁸⁵ Das INTERREG-IVB-Projekt betrifft eine wichtige Nord-Süd-Transversale und sollte Maßnahmen zur Beseitigung von infrastrukturellen Engpässen, zur Umweltentwicklung und zum Lärmschutz sowie zur Verlagerung von Güterverkehren von der Straße auf die Schiene anstoßen.⁸⁶ Der Korridor ist ein vorrangiges Projektvorhaben im Rahmen der von der Europäischen Union festgelegten Transeuropäischen Netze.⁸⁷

An dem Projektvorhaben CODE24 waren insgesamt 14 Projektpartner beteiligt.⁸⁸ Das multinationale Projekt zur Stärkung des europäischen Schienengüterverkehrs war auf eine Laufzeit von vier Jahren angelegt und wurde im Jahr 2015 beendet.⁸⁹

Der Korridor 24 selbst ist die Haupteisenbahnstrecke durch die Schweizer Alpen.⁹⁰ Gleichzeitig ist das Gebiet, auf dem der Korridor liegt, eines der dichtestbesiedelten und am stärksten industrialisierten Gebiete der Europäischen Union.⁹¹ Entscheidend für die angestrebte Steigerung des Transportpotentials innerhalb des Korridors und die langfristig angestrebte Verlagerung von Güterverkehren von der Straße auf die Schiene sind die Zulaufstrecken in Deutschland und Italien entscheidend, welche Logistikzentren nördlich und südlich der Alpen verbinden.⁹²

Der Korridor 24 ist eine wichtige Schienengüterstrecke, auf der jährlich ca. 35 Mrd. Tonnenkilometer an Gütern transport werden.⁹³ Diese Zahl soll sich Schätzungen zufolge bis

⁸³ Vgl. SCHECK/WILSKE (2011), S. 61.

⁸⁴ Weitere Informationen sind im Internet verfügbar. Vgl. CODE24 (2017).

⁸⁵ Vgl. SCHECK/WILSKE (2011), S. 57 f. u. 64.

⁸⁶ Vgl. SCHECK/WILSKE (2011).

⁸⁷ Im Rahmen der Transeuropäischen Netze sollen 30 derartige, vorrangige Projektvorhaben bis zum Jahr 2020 verwirklicht werden. Vgl. SCHECK/WILSKE (2011), S. 57 f.

⁸⁸ Vgl. BRUNS et al. (2010a), S. 27.

⁸⁹ Vgl. SCHECK/WILSKE (2011), S. 64 f.

⁹⁰ Vgl. SCHECK/WILSKE (2011), S. 58.

⁹¹ Vgl. SCHECK/WILSKE (2011), S. 58.

⁹² Vgl. SCHECK/WILSKE (2011), S. 58.

⁹³ Vgl. SCHECK/WILSKE (2011), S. 58.

zum Jahr 2020 verdoppeln.⁹⁴ Daher sollten im Rahmen des Projekts CODE24 die entlang des Korridors stattfindenden Güterverkehre gestärkt werden.⁹⁵ Gleichzeitig stand das Projekt vor vielfältigen Herausforderungen:⁹⁶ Es fehlten umfassende Informationen, wie viele Züge künftig den Korridor befahren werden.⁹⁷ Auch war unklar, wie hoch die zusätzlichen Kapazitäten sind, welche durch verbesserte Abläufe und Auslastung der bestehenden Infrastruktur genutzt werden können.⁹⁸ Dies war und ist von Bedeutung, da sich der Bau zusätzlicher Güterzugstrecken, im Vergleich zu bspw. dem Ausbau von Ladeterminals, schwierig gestaltet.⁹⁹ Denn wenn multimodale Transportdienstleistungen gefördert werden sollen, muss dies immer von einem Abbau von Zugangshürden sowie dem Aufbau spezifischen Know-hows begleitet werden.¹⁰⁰

Das Projekt CODE24 war in vier Arbeitspakete unterteilt, wobei jedes Arbeitspaket mehrere Teilprojekte umfasste.¹⁰¹ Im Rahmen des Arbeitspakets 3 – „Gütertransport und Logistik“ – wurde die Steigerung der Transportkapazität entlang des gesamten Korridors durch bessere wirtschaftliche Auslastung und Stärkung der Wettbewerbsfähigkeit des Verkehrsträgers der Schiene verfolgt.¹⁰² Weiterhin existierte im Rahmen des Arbeitspakets das Teilprojekt „Entwicklung einer Online-Güterbörse für Transportleistungen im Schienengüterverkehr“, welches sich mit der Entwicklung eines funktionsfähigen und stabilen Softwaresystems für eine internetbasierte Frachtenbörse für den Schienengüterverkehr beschäftigte.¹⁰³

Im folgenden Kapitel wird die Entwicklung des Softwareprototyps dieser „Online Rail Freight Exchange“ (ORFE) beschrieben. Es wird aufgezeigt, inwiefern die hierbei gewonnen Erkenntnisse für die Entwicklung einer über den Rahmen des CODE24-Projekts hinausgehenden Online-Frachtenbörse relevant sind.

⁹⁴ Vgl. SCHECK/WILSKE (2011), S. 58.

⁹⁵ Vgl. SCHECK/WILSKE (2011), S. 61.

⁹⁶ Im Folgenden wird auf jene primär die logistischen Zielsetzungen des Projekts betreffenden Herausforderungen eingegangen. Diese Herausforderungen bestanden jedoch auf allen Ebenen des Projekts. Vgl. SCHECK/WILSKE (2011), S. 61 ff.

⁹⁷ Vgl. SCHECK/WILSKE (2011), S. 63.

⁹⁸ Während ENDEMANN und KASPERS eine zusätzliche Kapazität von 20 bis 50 Prozent durch Reserven bei Zuggewicht und Zuglänge ermittelt haben, sehen es SCHECK und WILSKE als unklar, ob die bestehende Infrastruktur im Korridor noch Kapazitätsreserven bietet, welche durch Verbesserungen im Betriebsablauf aktiviert werden könnten. Vgl. ENDEMANN/KASPERS (2011), sowie SCHECK/WILSKE (2011), S. 63.

⁹⁹ Gründe für diese Schwierigkeit liegen in den hohen Zuggewichten, welche für Güterzüge eine – im Vergleich bspw. zum straßengebundenen Verkehr – Trassierung mit möglichst geringen Steigungen erforderlich machen. Dies erfordert wiederum kostenintensive Bauprojekte zur Realisierung von bspw. Tunneln und Brücken. Vgl. SCHECK/WILSKE (2011), S. 64.

¹⁰⁰ Vgl. SCHECK/WILSKE (2011), S. 63.

¹⁰¹ Die vier Arbeitspakete sind: Raumplanung und Infrastrukturentwicklung, Umwelt und Lärm, Gütertransport und Logistik sowie Kommunikation und Akzeptanz. Vgl. SCHECK/WILSKE (2011), S. 65 ff.

¹⁰² Vgl. SCHECK/WILSKE (2011), S. 67.

¹⁰³ Vgl. SCHECK/WILSKE (2011), S. 67.

2.4 Erkenntnisse zu Online-Frachtenbörsen im Rahmen des Projekts CODE24

2.4.1 Ziele der Etablierung

Durch die breite Verfügbarkeit von Internetanschlüssen sind nicht nur Unternehmen, sondern auch Privatpersonen in der Lage, Geschäfte auf elektronischem Wege abzuschließen. Dabei wurden klassische bilaterale Kommunikationsmittel wie Telefon oder Telefax zunächst durch die E-Mail-Technologie ergänzt. Allerdings konnten E-Mails nie den Geschäftsabschluss durch persönliche Unterschrift eines Vertrags ersetzen. Vielfach wurde ein Geschäft per Telefon oder E-Mail verhandelt, um anschließend per Telefax den unterschriebenen Vertrag auszutauschen.¹⁰⁴

Mit Aufkommen des Internets änderten sich die Umstände, unter denen Geschäfte angebahnt wurden. Es wurden Marktplätze geschaffen, in denen rechtsgültige Verträge ohne physische Unterschrift abgeschlossen werden können.¹⁰⁵ Hierdurch ermöglichten diese elektronischen Marktplätze erstmals, einen virtuellen Ort zu schaffen, an welchem Anbieter und Nachfrager zusammenkommen und Geschäfte abschließen können.¹⁰⁶ Diese Anbieter und Nachfrager – welche Teilnehmer eines elektronischen Marktplatzes und Benutzer der zur Teilnahme nötigen Software sind – können nun erstmals ortsungebunden Waren und Dienstleistungen handeln und zeitnah über den Erfolg oder Misserfolg ihrer Geschäftabschlüsse informiert werden.¹⁰⁷

Während die traditionelle Art der Geschäftsanbahnung im Schienengüterverkehr wie beschrieben die direkte Kommunikation zwischen Akteuren wie Verladern und Frachtführern darstellt, hat die Internettechnologie diesen Akteuren ebenfalls die beschriebenen neuartigen Kommunikationsmöglichkeiten zur Geschäftsanbahnung eröffnet. Allerdings nutzen die Akteure des Schienengüterverkehrs – Verloader, Frachtführer, Spediteure sowie Eisenbahnverkehrsunternehmen – weiterhin die traditionellen fernmündlichen Verfahren zur Geschäftsanbahnung.¹⁰⁸

ENDEMANN und KASPERS fassen die Ziele einer Online-Frachtenbörse im Rahmen des Projekts CODE24 zusammen:

¹⁰⁴ KOLLMANN spricht hier von „Medienbrüchen“. Vgl. KOLLMANN (2016), S. 152.

¹⁰⁵ GRIEGER stellt hierzu fest, dass das Konzept elektronischer Marktplätze bereits seit den 1940er Jahren existiert. Die Entwicklung dieser Marktplätze hätte jedoch erst mit Einsatz der Computer- und insbesondere der Internettechnologie entscheidende Fortschritte hin zur Dezentralisierung und Ubiquität gemacht. Vgl. GRIEGER (2003), S. 281.

¹⁰⁶ Vgl. GRIEGER (2003), S. 281 f.

¹⁰⁷ Vgl. GRIEGER (2003), S. 281 f.

¹⁰⁸ Gemeint sind hier primär Telefon und Telefax. Vgl. HABIB/BRUNS (2012), S. 1 u. 5.

Die Überwindung von Informationsdefiziten bei den Beteiligten steht im Vordergrund der Bemühungen.¹⁰⁹ Hinzu kommen die Unterstützung der Kooperation und Kommunikation bei Logistikern, Spediteuren und Eisenbahnverkehrsunternehmen sowie die Einbindung möglicher Lösungen für den Vor- und Nachlauf zur Schiene.¹¹⁰ Die Online-Frachtenbörse soll einen Beitrag zur effizienteren Abwicklung der Logistikkette und Reduzierung der Logistikkosten leisten und eine Verlagerung von Gütertransporten auf die Schiene begünstigen, um so die Reduktion der Lkw-Fahrten im Korridor 24 zu fördern.¹¹¹

Potentielle Nutzergruppen der Online-Frachtenbörse verbinden deren Etablierung mit gruppenindividuellen Zielen.

Aus Sicht der Eisenbahnverkehrsunternehmen ist das primäre Ziel der Etablierung einer Online-Frachtenbörse die Erlangung eines Mehrwerts durch ihre Nutzung.¹¹² Sie hoffen zusätzliche Geschäfte zu generieren, indem sie über die Online-Frachtenbörse Fracht zum Transport offerieren oder freien Laderaum anbieten.¹¹³ Die Chance des direkten Kundenkontakts wird ebenfalls angeführt.¹¹⁴ Weiterhin hoffen sie, Leerkapazitäten zu nutzen, um der Herausforderung zu begegnen, ihre Transportkapazitäten bis zur Kapazitätsgrenze auszulasten und leere Wagen – bspw. auf Rückfahrten – zu befüllen.¹¹⁵ Während dem Einkauf von Ressourcen beim Straßengüterverkehr nur die Rolle einer zusätzlichen, optionalen Leistung zugesprochen wird, würde sich die Möglichkeit bspw. Wagen über die Online-Frachtenbörse anzumieten als bedeutender Mehrwert für die Eisenbahnverkehrsunternehmen erweisen.¹¹⁶ Der Grund ist, dass sich der Ressourceneinkauf im Schienengüterverkehr im Vergleich zum Straßengüterverkehr als wesentlich komplexer darstellt.¹¹⁷ Die Eisenbahnverkehrsunternehmen hoffen, den Ressourceneinkauf über eine Online-Frachtenbörse erheblich vereinfachen zu können.¹¹⁸

Aus Sicht der Verlader ist das primäre Ziel ebenfalls die Erlangung eines Mehrwerts durch Nutzung einer Online-Frachtenbörse.¹¹⁹ Sie erhoffen sich diesen durch die Steigerung der Markttransparenz.¹²⁰ Die bessere Vergleichbarkeit von Preis- und Qualitätskriterien

¹⁰⁹ Vgl. ENDEMANN/KASPERS (2011), S. 33.

¹¹⁰ Vgl. ENDEMANN/KASPERS (2011), S. 33.

¹¹¹ Vgl. ENDEMANN/KASPERS (2011), S. 33.

¹¹² Vgl. KLIPPERT et al. (2011), S. 64.

¹¹³ Vgl. KLIPPERT et al. (2011), S. 64.

¹¹⁴ Vgl. KLIPPERT et al. (2011), S. 64.

¹¹⁵ Vgl. KLIPPERT et al. (2011), S. 64.

¹¹⁶ Vgl. KLIPPERT et al. (2011), S. 64.

¹¹⁷ Vgl. KLIPPERT et al. (2011), S. 65.

¹¹⁸ Vgl. KLIPPERT et al. (2011), S. 65.

¹¹⁹ Vgl. KLIPPERT et al. (2011), S. 65.

¹²⁰ Vgl. KLIPPERT et al. (2011), S. 65.

macht einen Vergleich nicht nur zwischen Dienstleistern, sondern auch zwischen verkehrsträgerübergreifenden Szenarien denkbar.¹²¹ So erscheint es möglich, dass ein Verlader erst durch einen solchen Vergleich erfährt, dass er einen geplanten Transport mit einem anderen Verkehrsträger oder einer Kombination aus Verkehrsträgern bspw. kostengünstiger oder schneller abwickeln könnte, als er es ohne Nutzung der Online-Frachtenbörse geplant hatte.¹²² So hoffen sie auch auf eine Verbesserung ihrer Prozesse durch Einsparungen von Prozesskosten und die Verkürzung von Prozessdurchlaufzeiten.¹²³ Weiterhin werden günstigere Einkaufspreise bei Transportdienstleistungen als Kriterium für das Erreichen eines Mehrwerts angeführt.¹²⁴

Die Ziele der Speditionen decken sich mit denen der vorgenannten, potentiellen Nutzergruppen.¹²⁵ Sie wollen einen Mehrwert aus der Steigerung der Markttransparenz, einer Verbesserung der Prozesse, günstigeren Einkaufspreisen sowie einem vereinfachten Ressourceneinkauf und der Nutzung von Leerkapazitäten erzielen.¹²⁶

Der Betreiber einer Online-Frachtenbörse muss seinerseits das Ziel verfolgen, seinen Gewinn zu maximieren.¹²⁷

Die Ziele von Eisenbahnverkehrsunternehmen, Verladern und Speditionen auf der einen und des Betreibers auf der anderen Seite offenbaren einen Konflikt.¹²⁸ Zum einen wird eine Online-Frachtenbörse für den Schienengüterverkehr gefordert und die Vergleichsmöglichkeit verschiedener Verkehrsträger und ihrer Kombination innerhalb eines Gesamttransports als Kriterium zur Schaffung eines Mehrwerts angegeben.¹²⁹ Zum anderen steht die Forderung nach einem breiten Leistungsspektrum der Online-Frachtenbörse seitens der potentiellen Nutzer dem Ziel der Gewinnmaximierung seitens des Betreibers gegenüber.¹³⁰ Während ein möglichst breites Leistungsspektrum notwendig ist, um die Ziele aller potentiellen Nutzergruppen zu erfüllen, bedarf die Realisierung dieses Leistungsspektrums hoher Investitionen seitens des Betreibers.¹³¹

¹²¹ Vgl. KLIPPERT et al. (2011), S. 65.

¹²² Vgl. KLIPPERT et al. (2011), S. 65.

¹²³ Vgl. KLIPPERT et al. (2011), S. 65.

¹²⁴ Vgl. KLIPPERT et al. (2011), S. 66.

¹²⁵ Vgl. KLIPPERT et al. (2011), S. 66.

¹²⁶ Vgl. KLIPPERT et al. (2011), S. 66.

¹²⁷ Vgl. KLIPPERT et al. (2011), S. 67.

¹²⁸ Vgl. KLIPPERT et al. (2011), S. 68.

¹²⁹ Vgl. KLIPPERT et al. (2011), S. 68.

¹³⁰ Vgl. KLIPPERT et al. (2011), S. 68.

¹³¹ Vgl. KLIPPERT et al. (2011), S. 68.

KLIPPERT et al. führen an, dass eventuell nicht alle Teilziele aller potentiellen Nutzergruppen abgedeckt werden müssen, um einzelnen Nutzergruppen einen wahrnehmbaren Mehrwert zu bieten.¹³²

Im folgenden Kapitel werden die in der bisherigen Forschung identifizierten Anforderungen an eine Online-Frachtenbörse zusammengefasst, um aus diesen Anforderungen und den Realproblemen die betriebswirtschaftlich wünschenswerten Eigenschaften von Online-Frachtenbörsen abzuleiten.

2.4.2 Anforderungen potentieller Nutzer

2.4.2.1 Prozesse und Sicherheit

Die Wichtigkeit der in den folgenden Kapiteln aufgeführten Anforderungen wird mittels der Verben „müssen“ und „sollen“ unterschieden. Verpflichtende Anforderungen werden mit dem Verb „müssen“, wünschenswerte Anforderungen mit dem Verb „sollen“ beschrieben.

Die Online-Frachtenbörse muss fähig sein, die Nutzer in den Informations- und Vereinbarungsphasen zu unterstützen.¹³³ Sie soll bei potentiellen Nutzern vorhandenen Prozess unterstützen, anstatt die Schaffung neuer Prozesse zu erfordern.¹³⁴ Hierzu soll den Nutzern die Möglichkeit geboten werden, Prozesse in der Online-Frachtenbörse schnell und einfach zu durchlaufen.¹³⁵ Mehrfache Dateneingaben sollen vermieden werden.¹³⁶ Hierzu soll die Möglichkeit gegeben sein, einmal eingegebene Daten wiederzuverwenden.¹³⁷

Die Online-Frachtenbörse muss aus Sicht der Nutzer einfach bedienbar sein und den Nutzern das Gefühl einer durchgehenden Kontrolle über die Prozesse vermitteln – eine „intelligente Online-Frachtenbörse“ wird explizit nicht gewünscht.¹³⁸

Übersichten mit sich häufig ändernden Datengrundlagen müssen automatisch aktualisiert werden.¹³⁹ Nutzer der Online-Frachtenbörse müssen automatisch benachrichtigt werden, wenn ein potentieller Geschäftspartner von der Online-Frachtenbörse identifiziert wird.¹⁴⁰

¹³² Vgl. KLIPPERT et al. (2011), S. 68.

¹³³ Vgl. KLIPPERT et al. (2011), S. 73.

¹³⁴ Vgl. KLIPPERT et al. (2011), S. 73.

¹³⁵ Vgl. KLIPPERT et al. (2011), S. 73.

¹³⁶ Vgl. KLIPPERT et al. (2011), S. 73.

¹³⁷ Vgl. KLIPPERT et al. (2011), S. 73.

¹³⁸ Vgl. KLIPPERT et al. (2011), S. 73.

¹³⁹ Vgl. KLIPPERT et al. (2011), S. 74.

¹⁴⁰ Vgl. KLIPPERT et al. (2011), S. 74.

Die Online-Frachtenbörse muss dem Nutzer hierfür eine Auswahl an Benachrichtigungsformen anbieten.¹⁴¹

Es muss eine hohe Datenqualität sichergestellt sein.¹⁴² Um eine hohe Datenqualität sicherzustellen, muss die Online-Frachtenbörse konsistente Daten vorhalten.¹⁴³ Formulare müssen so bereitgestellt werden, dass die Nutzer gezwungen sind detaillierte und konsistente Angaben zu machen.¹⁴⁴

Ein automatisches Einlesen von Daten aus den ERP- und WW-Systemen der Nutzer soll möglich sein, wenn diese Daten von der Online-Frachtenbörse verwendet werden können, um Werte in Formularen als Vorschlagswerte einzutragen.¹⁴⁵ Wenn die Online-Frachtenbörse ein automatisches Einlesen von Daten aus den ERP- und WW-Systemen der Nutzer ermöglicht, muss es den Nutzern die Möglichkeit bieten, die daraus resultierenden Vorschlagswerte zu ändern.¹⁴⁶

Die Online-Frachtenbörse soll in ihren Prozessen gut strukturiert und übersichtlich angeordnet sein.¹⁴⁷ Sie soll einfach in der Handhabung sein und hierzu Navigationshilfen bieten sowie möglichst alle Geschäftsprozesse automatisieren.¹⁴⁸ Zudem soll die Online-Frachtenbörse eine hohe Transparenz bieten, wie bspw. eine Nachverfolgungsmöglichkeit des gesamten Geschäftsvorgangs.¹⁴⁹

Sie soll eine Volltextsuche bieten und die Nutzer beim Inserieren¹⁵⁰ von Angeboten unterstützen.¹⁵¹ Es soll eine Online-Feedbackfunktion geben, um den inserierenden Nutzer eines Angebots bei Rückfragen kontaktieren zu können.¹⁵²

Weiterhin halten die Autoren fest, dass dem Datenschutz beim Betrieb einer Online-Frachtenbörse eine besondere Bedeutung zukommt.¹⁵³ Die Nutzer müssen sich sicher sein

¹⁴¹ Es werden verschiedene Benachrichtigungsformen vorgeschlagen. Es sollen ein akustisches Signal am PC, eine SMS, eine E-Mail und ein Fax zur Auswahl stehen. Die Online-Frachtenbörse muss mindestens eine Benachrichtigung per E-Mail ermöglichen. Vgl. KLIPPERT et al. (2011), S. 74.

¹⁴² Vgl. KLIPPERT et al. (2011), S. 74.

¹⁴³ Vgl. KLIPPERT et al. (2011), S. 74.

¹⁴⁴ Vgl. KLIPPERT et al. (2011), S. 74.

¹⁴⁵ Vgl. KLIPPERT et al. (2011), S. 74.

¹⁴⁶ Vgl. KLIPPERT et al. (2011), S. 74.

¹⁴⁷ Vgl. HABIB/BRUNS (2012), S. 24.

¹⁴⁸ Vgl. HABIB/BRUNS (2012), S. 25.

Es fällt auf, dass diese Forderung zumindest potentiell konfliktär zu der Aussage ist, dass explizit keine intelligente Online-Frachtenbörse gewünscht sei. Vgl. KLIPPERT et al. (2011), S. 73.

¹⁴⁹ Vgl. HABIB/BRUNS (2012), S. 36.

¹⁵⁰ Auf Basis der detailliert formulierten Anforderungen der potentiellen Stakeholder wurden beispielhafte Eingabemasken angefertigt. Vgl. HABIB/BRUNS (2012), S. 27 ff.

¹⁵¹ Vgl. HABIB/BRUNS (2012), S. 25.

¹⁵² Vgl. HABIB/BRUNS (2012), S. 35.

¹⁵³ Vgl. HABIB/BRUNS (2012), S. 36.

können, dass kein Dritter unbefugt Zugriff auf ihre Daten nehmen kann.¹⁵⁴ In diesem Zusammenhang spielt auch die Authentizität der Nutzer eine große Rolle, also die Sicherheit, dass die Glaubwürdigkeit aller Nutzer gewährt ist.¹⁵⁵ Vor dem Hintergrund nicht-persönlicher Geschäftsabschlüsse unter teils einander unbekannten Parteien erhält diese Anforderung besonderes Gewicht.¹⁵⁶

Um die Sicherheit der Plattform zu gewährleisten, wird vorgeschlagen, dass Eisenbahnverkehrsunternehmen Verlader aufgrund der mit ihnen gemachten geschäftlichen Erfahrungen bewerten können.¹⁵⁷ So soll ein Index in der Online-Frachtenbörse entstehen, welcher die Glaubwürdigkeit und Zuverlässigkeit der Verlader wiedergibt.¹⁵⁸ Weiterhin wird vorgeschlagen, dass nicht nur die Eisenbahnverkehrsunternehmen, sondern auch die Betreiber der Online-Frachtenbörse in diesen Index eingreifen können, um aufgrund von anderweitig akquirierten Informationen die Daten einzelner Verlader ändern zu können, wie bspw. Finanzlage oder Referenzen von Geschäftspartnern.¹⁵⁹ Abschließend wird darauf hingewiesen, dass auch von der Online-Frachtenbörse zu vergebende Zertifikate bei der Schaffung von Authentizität hilfreich sein können.¹⁶⁰ So können bspw. positive Bewertungen durch Geschäftspartner, eine Marktexistenz von mindestens drei Jahren sowie der Nachweis von Lizenzen und finanzieller Stabilität zur Folge haben, dass der betreffende Nutzer ein Zertifikat erhält.¹⁶¹

Im Rahmen der Sicherheit der Online-Frachtenbörse wird abschließend die Sichtbarkeit der Daten angesprochen.¹⁶² Die Nutzer müssen Einfluss darauf nehmen können, welche anderen Nutzer die von ihnen eingestellten Inhalte sehen können und welche nicht.¹⁶³ Insbesondere wird hervorgehoben, dass Anbieter für ihre Wettbewerber nicht sichtbar sein dürfen und alle Daten müssen von dem eingebenden Nutzer explizit für ihren jeweiligen Empfängerkreis freigegeben werden.¹⁶⁴ Jeder Nutzer solle zu diesem Zweck eigene Empfängerkreise definieren und so bestimmen können, wer zu seinen Kontakten gehört und nur diese anderen Nutzer dürfen von seiner Teilnahme an der Online-Frachtenbörse erfahren.¹⁶⁵

¹⁵⁴ Vgl. HABIB/BRUNS (2012), S. 36.

¹⁵⁵ Vgl. HABIB/BRUNS (2012), S. 36.

¹⁵⁶ Vgl. HABIB/BRUNS (2012), S. 36 f.

¹⁵⁷ Vgl. HABIB/BRUNS (2012), S. 37.

¹⁵⁸ Vgl. HABIB/BRUNS (2012), S. 37.

¹⁵⁹ Vgl. HABIB/BRUNS (2012), S. 37.

¹⁶⁰ Vgl. HABIB/BRUNS (2012), S. 37.

¹⁶¹ Vgl. HABIB/BRUNS (2012), S. 37.

¹⁶² Vgl. HABIB/BRUNS (2012), S. 38.

¹⁶³ Vgl. HABIB/BRUNS (2012), S. 38.

¹⁶⁴ Vgl. HABIB/BRUNS (2012), S. 38.

¹⁶⁵ Vgl. HABIB/BRUNS (2012), S. 38.

2.4.2.2 Technologie

Die Online-Frachtenbörse muss internetbasiert sein.¹⁶⁶ Sie muss sich aus einer Applikation und einer an diese Applikation angebundenen Datenbank zusammensetzen und so konzipiert sein, dass es dem Betreiber möglich ist, ASP- und Outsourcing-Lösungen zu nutzen.¹⁶⁷ Zudem soll sie mit einfachen Mitteln über einen PC am Arbeitsplatz nutzbar sein und dem Nutzer kein technisches Wissen abverlangen.¹⁶⁸ Es sollen nach Möglichkeit technische Hilfsmittel zur Verfügung stehen, welche Kontakte zwischen den Betreibern und Nutzern sowie zwischen den Nutzern untereinander ermöglichen.¹⁶⁹ Als Beispiel wird die Bereitstellung eines E-Mail- oder SMS-Service genannt.¹⁷⁰

Die Homepage der Online-Frachtenbörse muss einem marktspezifischen Design entsprechen.¹⁷¹ Die Online-Frachtenbörse muss stabil laufen und rund um die Uhr für die Nutzer verfügbar und für den Betreiber kontrollierbar sein.¹⁷² Die Benutzeroberfläche der Online-Frachtenbörse soll international ausgerichtet und in mehreren Sprachen verfügbar sein, so dass Nutzer die Online-Frachtenbörse in ihrer Landessprache nutzen können.¹⁷³ Sie soll möglichst nicht nur vom PC, sondern auch von mobilen Endgeräten, abrufbar sein.¹⁷⁴

Es muss sichergestellt sein, dass potentiellen Nutzern, welche eine Anbindung an die Online-Frachtenbörse wünschen, von den Mitarbeitern eine Schnittstellendefinition bereitgestellt werden kann.¹⁷⁵

Die verwendete Technologie soll sich an Standards, wie bspw. XML, orientieren.¹⁷⁶ Sie muss mindestens den Qualitätskriterien Wartbarkeit, Anpassbarkeit, Änderbarkeit, Fehlerresistenz und Sicherheit entsprechen.¹⁷⁷

¹⁶⁶ Vgl. HABIB/BRUNS (2012), S. 23.

¹⁶⁷ Vgl. KLIPPERT et al. (2011), S. 84.

¹⁶⁸ Vgl. HABIB/BRUNS (2012), S. 23.

¹⁶⁹ Vgl. HABIB/BRUNS (2012), S. 23.

¹⁷⁰ Vgl. HABIB/BRUNS (2012), S. 23.

¹⁷¹ Vgl. KLIPPERT et al. (2011), S. 85.

¹⁷² Vgl. HABIB/BRUNS (2012), S. 23, sowie KLIPPERT et al. (2011), S. 84.

¹⁷³ HABIB und BRUNS identifizieren Deutsch, Niederländisch und Italienisch als relevante Sprachen. Vgl. HABIB/BRUNS (2012), S. 35, sowie KLIPPERT et al. (2011), S. 81.

¹⁷⁴ Vgl. HABIB/BRUNS (2012), S. 23 u. 35.

¹⁷⁵ Vgl. KLIPPERT et al. (2011), S. 84.

¹⁷⁶ Vgl. KLIPPERT et al. (2011), S. 84.

¹⁷⁷ Vgl. KLIPPERT et al. (2011), S. 84.

2.4.2.3 Leistungen

Im Folgenden werden die Anforderungen der potentiellen Nutzer an die Kernleistungen einer Online-Frachtenbörse für den Schienengüterverkehr beschrieben, welche zur Zusammenführung von Angebot und Nachfrage notwendig sind und direkt der Vermittlung von Transportdienstleistungen dienen. Anschließend werden die Anforderungen an zusätzliche Leistungen beschrieben, welche die Kernleistungen ergänzen.

Bereits bei den Kernleistungen wird jedoch deutlich, dass der gemachte Anforderungskatalog aus konzeptioneller Sicht auf die Schaffung einer multimodalen Frachtenbörse abzielt, welche anschließend vor dem Hintergrund der politischen Zielsetzungen des Projekts CODE24 nur für den kombinierten Verkehr genutzt werden soll.

Die Anforderungen an Kernleistungen sind:

Die Online-Frachtenbörse muss die neutrale Inserierung von Frachtvolumina zur Suche geeigneter Dienstleister für die Durchführung von Transporten unterstützen und für die Nachfrager die Ausschreibung von Transporten unterstützen.¹⁷⁸ Sie muss den Nutzern eine umfangreiche Suche ermöglichen¹⁷⁹ und die Aktualität der Inhalte – insbesondere der Inserate – gewährleisten.¹⁸⁰

Neben dem konventionellen Schienengüterverkehr muss die Online-Frachtenbörse auch den kombinierten Verkehr abdecken.¹⁸¹ Der Vor- und Nachlauf per Lkw¹⁸² und gegebenenfalls die Infrastrukturbetreiber von KV-Terminals sind somit einzubeziehen.¹⁸³ Es muss den Nutzern möglich sein, sowohl Fracht- als auch Laderaum für Einzelwagen, Wagengruppen und Ganzzüge einzustellen.¹⁸⁴ Es muss eine interaktive Kommunikation ermöglicht und die Plattform regelmäßig gepflegt werden.¹⁸⁵

Die Online-Frachtenbörse muss den Kontraktmarkt und soll den Spotmarkt abdecken.¹⁸⁶ Weiterhin muss der Transport- und der Ressourcenmarkt abgedeckt werden und es muss den Nutzern möglich sein, Wagen und Zugkapazitäten anzubieten.¹⁸⁷ Sie soll Möglichkeiten zur Nutzung des Spotmarktes und zur Auslastung vorhandener Restkapazitäten

¹⁷⁸ Vgl. ENDEMANN/KASPERS (2011), S. 35, sowie KLIPPERT et al. (2011), S. 74.

¹⁷⁹ Vgl. KLIPPERT et al. (2011), S. 74.

¹⁸⁰ Vgl. HABIB/BRUNS (2012), S. 42 u. 44, sowie KLIPPERT et al. (2011), S. 75.

¹⁸¹ Vgl. KLIPPERT et al. (2011), S. 75.

¹⁸² Vgl. HABIB/BRUNS (2012), S. 26 u. 29, sowie ENDEMANN/KASPERS (2011), S. 32.

¹⁸³ Vgl. HABIB/BRUNS (2012), S. 42, sowie ENDEMANN/KASPERS (2011), S. 32 u. 35.

¹⁸⁴ Vgl. KLIPPERT et al. (2011), S. 75.

¹⁸⁵ Vgl. ENDEMANN/KASPERS (2011), S. 35.

¹⁸⁶ Wenn die Online-Frachtenbörse den Spotmarkt abdeckt, muss es den Nutzern möglich sein, Nachfragen nach und Angebote für Stückgutverkehre und Einzelwagen einzustellen. Vgl. HABIB/BRUNS (2012), S. 41, sowie KLIPPERT et al. (2011), S. 75.

¹⁸⁷ Vgl. HABIB/BRUNS (2012), S. 42, sowie KLIPPERT et al. (2011), S. 75.

bieten sowie den Aufbau neuer Verkehre unterstützen.¹⁸⁸ Die Chance zur Akquisition von Neuverkehren über eine Online-Frachtenbörse wird insbesondere von den Anbietern von Transportdienstleistungen hervorgehoben.¹⁸⁹

Es muss für Wagen die Suche nach Rückladungen und für freie Zugkapazitäten die Suche nach Wagen möglich sein.¹⁹⁰ Umgekehrt muss auch für Rückladungen die Suche nach freien Wagen und für Wagen die Suche nach freier Zugkapazität möglich sein.¹⁹¹

Es muss den potentiellen Nutzergruppen Verlader, Speditionen und Frachtführer möglich sein, die Online-Frachtenbörse zu nutzen.¹⁹² Zusätzlich soll weiteren potentiellen Nutzergruppen die Nutzung der Online-Frachtenbörse möglich sein, bspw. Dienstleistern, Terminalbetreibern, Umschlagbetrieben sowie Wagen- und Lokomotiveneigentümern.¹⁹³ Die Online-Frachtenbörse soll keine Einschränkung auf bestimmte Nutzergruppen vorgeben.¹⁹⁴

Die Online-Frachtenbörse soll den potentiellen Nutzern einen Vergleich unterschiedlicher Verkehrsträger und unterschiedlicher Kombinationen dieser Verkehrsträger ermöglichen.¹⁹⁵ Wenn den Nutzern ein Vergleich unterschiedlicher Verkehrsträger angeboten wird, muss die Online-Frachtenbörse mindestens die Verkehrsträger Bahn, Lkw und Binnenschiff integrieren.¹⁹⁶

Es muss den Nutzern möglich sein, potentielle Geschäftspartner zu finden.¹⁹⁷

Es muss den Nutzern möglich sein, den konkreten Preis für die Dienstleistungen außerhalb des Systems auszuhandeln, Geschäftspartner nicht nur anhand des Preises auswählen zu müssen sowie zeitkritische Anfragen und Angebote durchzuführen.¹⁹⁸ Für langfristige Kontrakte muss die Online-Frachtenbörse den Nutzern Ausschreibungen ermöglichen.¹⁹⁹ Es muss den ausschreibenden Nutzern möglich sein, einen Höchstpreis für die ausgeschriebene Transportdienstleistung anzugeben.²⁰⁰

¹⁸⁸ Vgl. ENDEMANN/KASPERS (2011), S. 35.

¹⁸⁹ Vgl. ENDEMANN/KASPERS (2011), S. 35.

¹⁹⁰ Vgl. KLIPPERT et al. (2011), S. 75.

¹⁹¹ Vgl. KLIPPERT et al. (2011), S. 75.

¹⁹² Vgl. KLIPPERT et al. (2011), S. 76.

¹⁹³ Vgl. KLIPPERT et al. (2011), S. 76.

¹⁹⁴ Vgl. KLIPPERT et al. (2011), S. 76.

¹⁹⁵ Vgl. KLIPPERT et al. (2011), S. 76.

¹⁹⁶ Vgl. KLIPPERT et al. (2011), S. 76.

¹⁹⁷ Vgl. KLIPPERT et al. (2011), S. 76.

¹⁹⁸ Vgl. KLIPPERT et al. (2011), S. 76 f.

¹⁹⁹ Vgl. KLIPPERT et al. (2011), S. 77.

²⁰⁰ Vgl. KLIPPERT et al. (2011), S. 77.

Es muss den Nutzern möglich sein, ein ausführliches Tätigkeitsprofil in der Online-Frachtenbörse zu hinterlegen, um sich gegenüber anderen Nutzern zu präsentieren.²⁰¹ Es muss sichergestellt sein, dass die Angaben aus dem Tätigkeitsprofil von ausreichend hoher Detailliertheit und Konsistenz sind, damit sie die Grundlage für Matchings bilden können, sobald eine Auswahl an potentiellen Geschäftspartnern gesucht wird.²⁰²

Es sollen potentielle Geschäftspartner als Vorauswahl aufgelistet werden und dem ausschreibenden Nutzer soll die Möglichkeit geboten werden, unerwünschte Geschäftspartner manuell zu deselektieren.²⁰³ Es muss sichergestellt sein, dass einem ausschreibenden Nutzer nur Dienstleister als potentielle Geschäftspartner vorgeschlagen werden, die ein echtes Interesse an einem entsprechenden Auftrag haben.²⁰⁴

Informationen über die Nutzer sollen nur den Nutzern zur Verfügung stehen und nicht von Dritten eingesehen werden können.²⁰⁵ Die Ausschreibungen, Nachfragen und Angebote sollen nur von den Nutzern eingesehen werden können, welche vom ausschreibenden Nutzer ausgewählt wurden.²⁰⁶ Potentielle Geschäftspartner sollen bei Ausschreibungen, Nachfragen und Angeboten nicht anonymisiert sein.²⁰⁷

Nutzer, welche geschlossene Benutzergruppen erstellen und nutzen, sollen von neu registrierten Nutzern erfahren, welche als potentielle Geschäftspartner in Frage kommen.²⁰⁸

Es soll den Nutzern die Auswahl von potentiellen Geschäftspartnern erleichtert werden, indem Informationen zur Bonität der Nutzer in der Online-Frachtenbörse hinterlegt werden, welche von anderen Nutzern abgefragt werden können.²⁰⁹

Es soll Geschäftspartnern möglich sein, den jeweils anderen Geschäftspartner in der Online-Frachtenbörse zu bewerten.²¹⁰ Ein Geschäftspartner soll jedoch keine ungerechtfertigte Bewertung für den jeweils anderen Geschäftspartner in der Online-Frachtenbörse eintragen können.²¹¹

²⁰¹ Vgl. KLIPPERT et al. (2011), S. 79.

²⁰² Vgl. KLIPPERT et al. (2011), S. 79.

²⁰³ Vgl. KLIPPERT et al. (2011), S. 79.

²⁰⁴ Vgl. KLIPPERT et al. (2011), S. 79.

²⁰⁵ Vgl. KLIPPERT et al. (2011), S. 79.

²⁰⁶ Vgl. KLIPPERT et al. (2011), S. 79.

²⁰⁷ Vgl. KLIPPERT et al. (2011), S. 79.

²⁰⁸ Vgl. KLIPPERT et al. (2011), S. 80.

²⁰⁹ Vgl. KLIPPERT et al. (2011), S. 80.

²¹⁰ Vgl. KLIPPERT et al. (2011), S. 80.

²¹¹ Vgl. KLIPPERT et al. (2011), S. 80.

Es soll den Nutzern möglich sein, erstellte Nachfragen oder Angebote zu einem vorgegebenen Zeitpunkt zu versenden sowie nicht mehr aktuelle Nachfragen und Angebote zu löschen.²¹²

Es soll den Nutzern möglich sein, Stammdaten zu verändern.²¹³ Auch muss es den Nutzern möglich sein, für einzelne Mitarbeiter ihres Unternehmens neue Nutzer anzulegen, von ihnen angelegte Nutzer zu einem späteren Zeitpunkt zu löschen sowie die Rechtezuweisung für einzelne Nutzer selbst zu steuern.²¹⁴

Es soll den Nutzern möglich sein, zu den Aktivitäten eines anderen Nutzers in der Online-Frachtenbörse eine Übersicht aufzurufen.²¹⁵ Es sollen für die Nutzer Brancheninformationen bereitgestellt werden, welche unmittelbar aus den Aktivitäten der Online-Frachtenbörse abgeleitet werden.²¹⁶

Die Online-Frachtenbörse muss europaweit ausgerichtet sein, um internationale Verkehre zu unterstützen²¹⁷ und eine breite geographische Abdeckung zu gewährleisten.²¹⁸

Es muss den Nutzern möglich sein, verkehrsträgerspezifische Kartenansichten zu nutzen.²¹⁹

Es muss den Nutzern möglich sein, sich kritisch zur Online-Frachtenbörse zu äußern.²²⁰

Die Teilnehmer sollen eine hohe Aktivität in der Online-Frachtenbörse wahrnehmen.²²¹

Zudem soll die Online-Frachtenbörse den kombinierten Verkehr berücksichtigen.²²² Da viele Verlader nicht über einen eigenen Gleisanschluss verfügen und infolgedessen den Vor- und Nachlauf per Lkw organisieren müssen, soll die Online-Frachtenbörse ebenfalls Güterverkehre für den Verkehrsträger Straße anbieten.²²³ Allerdings sollen die hierbei abschließbaren Verträge auf den Vor- und Nachlauf zur Schiene begrenzt sein, um dem Teilziel „Verlagerung von Güterverkehren von der Straße auf die Schiene“ des Projekts CODE24 Rechnung zu tragen.²²⁴

²¹² Vgl. KLIPPERT et al. (2011), S. 80.

²¹³ Vgl. KLIPPERT et al. (2011), S. 81.

²¹⁴ Vgl. KLIPPERT et al. (2011), S. 81.

²¹⁵ Vgl. KLIPPERT et al. (2011), S. 81.

²¹⁶ Vgl. KLIPPERT et al. (2011), S. 81.

²¹⁷ Vgl. HABIB/BRUNS (2012), S. 35, ENDEMANN/KASPERS (2011), S. 35, sowie KLIPPERT et al. (2011), S. 81.

²¹⁸ Vgl. KLIPPERT et al. (2011), S. 81.

²¹⁹ Vgl. KLIPPERT et al. (2011), S. 81.

²²⁰ Vgl. KLIPPERT et al. (2011), S. 81.

²²¹ Vgl. KLIPPERT et al. (2011), S. 81.

²²² Vgl. HABIB/BRUNS (2012), S. 41.

²²³ Vgl. HABIB/BRUNS (2012), S. 41.

²²⁴ Vgl. HABIB/BRUNS (2012), S. 41.

Die potentiellen Nutzer regen an, dass die Online-Frachtenbörse Funktionen bereitstellen solle, über welche Eisenbahnverkehrsunternehmen andere Eisenbahnverkehrsunternehmen beauftragen können, für sie einen Gütertransport durchzuführen.²²⁵

Weiterhin soll die Online-Frachtenbörse die Buchung von KV-Terminals ermöglichen.²²⁶ Dabei soll sie Informationen bereitstellen, zu welchen Zeiten die gewünschten Terminals bereits belegt und zu welchen sie noch verfügbar sind. Anschließend soll eine Reservierung des Terminals über das System der Online-Frachtenbörse möglich sein.²²⁷

Außerdem soll die Buchung von Transporten und Zusatzdienstleistungen, wie bspw. die Anmietung von Wagen oder Personal, über die Online-Frachtenbörse möglich sein.²²⁸

Die Nutzung der Online-Frachtenbörse soll den Nutzern einen Mehrwert bieten.²²⁹ Hierzu sei es notwendig, dass sie darauf zielen müsse, die Möglichkeit zu bieten, neue Eisenbahngüterverkehre zur Verfügung zu stellen, und vermeiden solle, einen Marktplatz für Restkapazitäten²³⁰ darzustellen.²³¹

Weiterhin sollen Nutzer, deren Anfrage keine Ergebnisse liefert, eine Liste von möglichen Alternativprodukten angezeigt bekommen.²³² Zusätzlich können den Nutzern bei jeder Anmeldung im System der Online-Frachtenbörse initiativ Angebote angezeigt werden, welche für sie von Interesse sein könnten.²³³ Es wird weiterhin angeregt, dass sich im virtuellen Umfeld der Online-Frachtenbörse Communities bilden können, in denen sich die Nutzer untereinander austauschen.²³⁴

Die Anforderungen an zusätzliche Leistungen sind:

Die Online-Frachtenbörse muss den Nutzern den Handel weiterer verkehrsträgerspezifischer Leistungen und Ressourcen ermöglichen, wie bspw. die Anmietung von Wagen durch eine Integration von Wagenvermietern in das Angebot der Online-Frachtenbörse und die Vermittlung von Personaldienstleistungen.²³⁵

²²⁵ Vgl. HABIB/BRUNS (2012), S. 42.

²²⁶ Vgl. HABIB/BRUNS (2012), S. 42.

²²⁷ Vgl. HABIB/BRUNS (2012), S. 42.

²²⁸ Vgl. HABIB/BRUNS (2012), S. 42.

²²⁹ Vgl. HABIB/BRUNS (2012), S. 42.

²³⁰ Bzgl. der Abwicklung von Restkapazitäten finden sich in der Literatur widersprüchliche Aussagen. Vgl. ENDEMANN/KASPERS (2011), S. 35.

²³¹ Vgl. HABIB/BRUNS (2012), S. 42.

²³² Vgl. HABIB/BRUNS (2012), S. 43.

²³³ Vgl. HABIB/BRUNS (2012), S. 43.

²³⁴ Vgl. HABIB/BRUNS (2012), S. 43.

²³⁵ Vgl. ENDEMANN/KASPERS (2011), S. 35, sowie KLIPPERT et al. (2011), S. 67 u. 76.

Die Online-Frachtenbörse soll transparent sein und den Nutzern auf diese Weise eine bessere Vergleichbarkeit von angebotenen Dienstleistungen sowie deren Preisen bieten.²³⁶ Sie soll kapazitätsspezifische Informationen zu bereits existierenden Transportströmen und Gleisanschlüssen bereitstellen sowie eine Ladestellensuche über die Anlagen der DB Netz AG und anderer Netzbetreiber ermöglichen.²³⁷ Hierdurch sollen der Informationsaustausch, die Kommunikation und die mögliche Kooperation zwischen potentiellen Partnern innerhalb des Schienengüterverkehrs gefördert werden.²³⁸ In diesem Zusammenhang wird auch eine Bereitstellung aller Inhalte der Online-Frachtenbörse zum Aufruf per Mobiltelefon gefordert.²³⁹ Auf diesem Wege sollen die bestehenden Informationsbarrieren und -asymmetrien im Schienengüterverkehrsmarkt überwunden werden.²⁴⁰

Außerdem soll sie den Anbietern innerhalb der Frachtenbörse Informationen von Verladern über potentielle Transportmengen (für den Aufbau neuer Züge) sowie zur Identifikation von Paarigkeiten zur Verfügung stellen.²⁴¹

Zudem sollen alle Sonderbedingungen und möglichen Einschränkungen sowohl auf der Eisenbahnstrecke als auch auf der Strecke des Vor- und Nachlaufs per Lkw mit Hilfe der Online-Frachtenbörse ersichtlich sein.²⁴² So sollen bspw. Verkehrsunfälle und Baustellen auf der Strecke des Transports zeitnah im System der Online-Frachtenbörse auftauchen.²⁴³

Der Kundendienst der Online-Frachtenbörse soll zu festen Zeiten – bestenfalls rund um die Uhr – schnell erreichbar sein, wenn Nutzer Rückfragen zu Transporten haben.²⁴⁴ Der Kundendienst soll per Telefon und E-Mail erreichbar sein.²⁴⁵ Zudem soll die Online-Frachtenbörse über ein separates Bewertungssystem für den Kundendienst verfügen, so dass sich potentielle Nutzer ein Bild über die Zufriedenheit der bestehenden Nutzer in Bezug auf die Koordination der Geschäftsabwicklung seitens der Online-Frachtenbörse machen können.²⁴⁶ Zusätzlich soll ein Beschwerdemanagement zur Verfügung stehen, da Bewertungen und Beschwerden nach Ansicht der potentiellen Nutzer auf diese Weise zu einer Verbesserung der Online-Frachtenbörse beitragen werden.²⁴⁷

²³⁶ Vgl. HABIB/BRUNS (2012), S. 35 f.

²³⁷ Vgl. ENDEMANN/KASPERS (2011), S. 32 u. 35.

²³⁸ Vgl. ENDEMANN/KASPERS (2011), S. 35.

²³⁹ Vgl. HABIB/BRUNS (2012), S. 35.

²⁴⁰ Vgl. ENDEMANN/KASPERS (2011), S. 35.

²⁴¹ Vgl. ENDEMANN/KASPERS (2011), S. 35.

²⁴² Vgl. HABIB/BRUNS (2012), S. 44.

²⁴³ Vgl. HABIB/BRUNS (2012), S. 44.

²⁴⁴ Vgl. HABIB/BRUNS (2012), S. 45.

²⁴⁵ Vgl. HABIB/BRUNS (2012), S. 45.

²⁴⁶ Vgl. HABIB/BRUNS (2012), S. 45.

²⁴⁷ Vgl. HABIB/BRUNS (2012), S. 45.

2.4.2.4 Betreiber

Aus den vorangegangenen Überlegungen zu Sicherheitsaspekten²⁴⁸ einer Online-Frachtenbörsen ergeben sich die folgende Anforderungen hinsichtlich der (natürlichen oder institutionellen) Person des Betreibers einer Online-Frachtenbörse.

Eine Online-Frachtenbörse sollte von einer neutralen Partei betrieben werden, welche keine eigenen Geschäftsinteressen im Schienengüterverkehr besitzt.²⁴⁹ Die Forderung nach Neutralität erstreckt sich hierbei auf alle Mitarbeiter der Online-Frachtenbörse.²⁵⁰

Die internen Bereiche Marketing, Vertrieb, Controlling, Administration und Leitung der Online-Frachtenbörse sollten bei ihren Arbeiten durch das Design der Online-Frachtenbörse unterstützt werden.²⁵¹ Die Online-Frachtenbörse sollte für diese Bereiche vordefinierte Rollen anbieten.²⁵² Prozesse, welche die Arbeit der Mitarbeiter der Online-Frachtenbörse unterstützen, sollten möglichst vollständig automatisiert ablaufen.²⁵³ Zudem muss gewährleistet sein, dass die von den Mitarbeitern der Online-Frachtenbörse verwendeten Benutzeroberflächen ein hohes Maß an Usability aufweisen, um die Arbeit der Mitarbeiter zu erleichtern.²⁵⁴

Die Eingabemasken für Telesales-Mitarbeiter sollten nicht mehr Informationen anzeigen, als für die Wahrnehmung der Aufgaben der jeweiligen Mitarbeiter notwendig sind.²⁵⁵

Es muss eine Ansicht existieren, welcher die Mitarbeiter entnehmen können, in welchem Bearbeitungszustand ein neu einzurichtender Nutzer ist.²⁵⁶

Der für einen Nutzer zuständige Mitarbeiter muss von der Online-Frachtenbörse automatisch benachrichtigt werden, wenn dieser Nutzer von seinem üblichen Nutzungsverhalten abweicht.²⁵⁷

Den Mitarbeitern der Online-Frachtenbörse muss sowohl die Erfassung der aktiven Nutzer als auch potentieller Nutzer möglich sein.²⁵⁸

²⁴⁸ Vgl. Kapitel 2.4.2.1, S. 18.

²⁴⁹ Vgl. HABIB/BRUNS (2012), S. 38.

²⁵⁰ Vgl. HABIB/BRUNS (2012), S. 38.

²⁵¹ Vgl. KLIPPERT et al. (2011), S. 81.

²⁵² Vgl. KLIPPERT et al. (2011), S. 81.

²⁵³ Vgl. KLIPPERT et al. (2011), S. 82.

²⁵⁴ Vgl. KLIPPERT et al. (2011), S. 82.

²⁵⁵ Vgl. KLIPPERT et al. (2011), S. 82.

²⁵⁶ Vgl. KLIPPERT et al. (2011), S. 82.

²⁵⁷ Vgl. KLIPPERT et al. (2011), S. 82.

²⁵⁸ Vgl. KLIPPERT et al. (2011), S. 82.

Den Mitarbeitern muss es möglich sein, die Daten jedes Nutzers zu ändern und Nutzer zu löschen.²⁵⁹ Nutzer, die gekündigt haben, müssen zum 1. eines vorgegebenen Monats um 0 Uhr automatisch in der Online-Frachtenbörse deaktiviert oder gelöscht werden.²⁶⁰

Die Mitarbeiter müssen in der Lage sein, einzelne User-ID für Nutzer anzulegen, zu aktivieren und zu deaktivieren.²⁶¹ Zusätzlich muss es den Mitarbeitern möglich sein, einer User-ID Rechte und Rollen zuzuweisen.²⁶²

Die Mitarbeiter müssen in der Lage sein, Nutzer in A-, B- und C-Kunden zu unterteilen.²⁶³

Den Mitarbeitern müssen analytische Übersichten zur Verfügung stehen, welche nach Teilnehmern aufgeschlüsselt anzeigen können, was diese an einem bestimmten Tag oder im laufenden Monat des aktuellen Jahres gemacht haben.²⁶⁴ Den Mitarbeitern der Online-Frachtenbörse müssen Listen aller Nutzer zur Verfügung stehen, welche den Mitarbeitern ermöglichen, sich die Entwicklung des Kundenstamms anzeigen zu lassen.²⁶⁵ In diesem Zusammenhang müssen auch Hochrechnungen möglich sein.²⁶⁶ Weiterhin muss es den Mitarbeitern möglich sein, in Zukunft auch Abfragen zu erstellen, für die zurzeit kein Bedarf besteht.²⁶⁷

Dem Betreiber der Online-Frachtenbörse muss es möglich sein, mit Hilfe von Übersichten Umsatzprognosen zu erstellen und Quartalsergebnisse zu vergleichen.²⁶⁸ Weiterhin muss es dem Betreiber möglich sein, die Aktivitäten der Mitarbeiter zu verfolgen.²⁶⁹

Dem Vertrieb müssen Auswertungen zur Verfügung stehen, welche aufzeigen, inwieweit Kontakte zu potentiellen und tatsächlichen Nutzern hergestellt wurden und wie viele Verträge abgeschlossen wurden.²⁷⁰

²⁵⁹ Vgl. KLIPPERT et al. (2011), S. 82.

²⁶⁰ Vgl. KLIPPERT et al. (2011), S. 82.

²⁶¹ Vgl. KLIPPERT et al. (2011), S. 82 f.

²⁶² Vgl. KLIPPERT et al. (2011), S. 83.

²⁶³ Vgl. KLIPPERT et al. (2011), S. 83.

²⁶⁴ Vgl. KLIPPERT et al. (2011), S. 83.

²⁶⁵ Vgl. KLIPPERT et al. (2011), S. 82.

²⁶⁶ Vgl. KLIPPERT et al. (2011), S. 83.

²⁶⁷ Vgl. KLIPPERT et al. (2011), S. 84.

²⁶⁸ Vgl. KLIPPERT et al. (2011), S. 83.

²⁶⁹ Vgl. KLIPPERT et al. (2011), S. 84.

²⁷⁰ Vgl. KLIPPERT et al. (2011), S. 83.

Es muss sichergestellt sein, dass in der Online-Frachtenbörse durchführbare Auswertungen auch grafische Abbildungen enthalten.²⁷¹ Dem Marketing und dem Vertrieb muss es möglich sein, Kampagnen durchzuführen und auszuwerten.²⁷²

Die Gebührenmodelle für Nutzer müssen von den Mitarbeitern innerhalb der Online-Frachtenbörse so flexibel gestaltbar sein, dass kostenfreie Probemonate, Rabattaktionen und kundenindividuelle Gebührenmodelle eingepflegt werden können.²⁷³

Es muss sichergestellt sein, dass die Startseite und weitere Informationsseiten der Homepage von den Mitarbeitern der Online-Frachtenbörse ohne besondere technische Kenntnisse – bspw. HTML-Kenntnisse – erstellt und aktualisiert werden können.²⁷⁴

2.4.2.5 Finanzen

Neben einem neutralen Betreiber erfordert das Betreiben einer Online-Frachtenbörse zudem ein langfristig tragfähiges Geschäftsmodell.

Da die Anzahl der beteiligten Akteure im Schienengüterverkehr geringer ist als im Straßengüterverkehr, erscheint eine Umlage der anfallenden Kosten zum Betrieb der Online-Frachtenbörse schwierig, da hierbei wenige Teilnehmer die Kosten tragen müssten.²⁷⁵ So kommen die von HABIB und BRUNS befragten potentiellen Nutzer zu dem Schluss, dass weder die Verlader noch die Nachfrager von Transportdienstleistungen für die Inanspruchnahme der Leistungen der Online-Frachtenbörse zahlen sollten, sondern die Anbieter.²⁷⁶ Des Weiteren sollten Eisenbahnverkehrsunternehmen eine Kostenübersicht ihrer Leistungen auf der Online-Frachtenbörse veröffentlichen, um so eine höhere Transparenz und Vergleichbarkeit mit Straßengüterverkehren zu erreichen.²⁷⁷

Daher werden verschiedene Modelle diskutiert, die den kostendeckenden Betrieb einer Online-Frachtenbörse möglich machen sollen.²⁷⁸ So bestehe die Möglichkeit eine Grundgebühr, eine Anmeldegebühr und eine Transaktionsgebühr zu erheben.²⁷⁹ Die Online-Frachtenbörse muss so entwickelt werden, dass die Nutzer die Kosten der Software als akzeptabel wahrnehmen.²⁸⁰ Zusätzlich wird vorgeschlagen, auf den Seiten der Online-

²⁷¹ Vgl. KLIPPERT et al. (2011), S. 83.

²⁷² Vgl. KLIPPERT et al. (2011), S. 83.

²⁷³ Vgl. KLIPPERT et al. (2011), S. 83.

²⁷⁴ Vgl. KLIPPERT et al. (2011), S. 83.

²⁷⁵ Vgl. HABIB/BRUNS (2012), S. 39.

²⁷⁶ Vgl. HABIB/BRUNS (2012), S. 39 f.

²⁷⁷ Vgl. HABIB/BRUNS (2012), S. 40.

²⁷⁸ Vgl. HABIB/BRUNS (2012), S. 40.

²⁷⁹ Vgl. HABIB/BRUNS (2012), S. 40.

²⁸⁰ Vgl. HABIB/BRUNS (2012), S. 40, sowie KLIPPERT et al. (2011), S. 85.

Frachtenbörse Werbung zu schalten, um so einen Teil der Betriebskosten zu decken.²⁸¹

2.4.3 Analyse des kompetitiven Umfelds

Durch die hohe Bedeutung der Schiene als Verkehrsträger paneuropäischer Güterverkehre steigt die Notwendigkeit zur Gründung einer Online-Frachtenbörse zur Reduzierung der bestehenden Probleme und Verbesserung der Güterverkehre auf diesem Verkehrsträger.²⁸² Da jedoch aufgrund des – im Vergleich zum straßengebundenen Güterverkehr – relativ kleinen Marktumfelds nur wenige Akteure als Teilnehmer für eine solche Online-Frachtenbörse in Frage kommen, gestaltet sich die Planung schwierig. Vielen erscheint es fraglich, inwieweit eine Online-Frachtenbörse für ihre – vielfach auf persönlichem Kontakt und Vertrauen beruhenden, seit Jahren bestehenden – Geschäftsbeziehungen förderlich sein kann.²⁸³

HABIB und BRUNS stellen im Rahmen ihrer Literaturrecherche gefundene Kriterien für die erfolgreichen Etablierung einer Online-Frachtenbörse zusammen.²⁸⁴ Die wichtigsten Kriterien, damit sich potentielle Nutzer für eine Online-Frachtenbörse entscheiden, sind demnach:²⁸⁵

- Es muss eine Anzahl von Frachten/Laderäumen im System der Online-Frachtenbörse bestehen.
- Es muss eine Anzahl an Kunden bestehen, die sich vertraglich an die Online-Frachtenbörse gebunden haben.
- Die Online-Frachtenbörse muss von den Nutzern in der eigenen Landessprache genutzt werden können.
- Die Online-Frachtenbörse muss in Ländern außerhalb Deutschlands eingeführt werden.
- Die Online-Frachtenbörse muss in diesen Ländern von einer Anzahl an Kunden genutzt werden.
- Die Online-Frachtenbörse muss eine Bonitätsprüfung der Nutzer sicherstellen.
- Die Nutzer müssen die Möglichkeit haben, eine möglichst kostenlose Kundendienst-Hotline zu nutzen.
- Die Zugangsvoraussetzungen müssen aus Sicht der potentiellen Nutzer einfach sein.

²⁸¹ Vgl. HABIB/BRUNS (2012), S. 40 f.

²⁸² Vgl. HABIB/BRUNS (2012), S. 11.

²⁸³ Vgl. HABIB/BRUNS (2012), S. 11.

²⁸⁴ Vgl. HABIB/BRUNS (2012), S. 13 f.

²⁸⁵ Vgl. HABIB/BRUNS (2012), S. 13 f.

- Die potentiellen Nutzer müssen eine Übersicht über einmalige und regelmäßige Kosten, die bei der Nutzung der Online-Frachtenbörse anfallen, abrufen können.
- Die Nutzer müssen die Möglichkeit erhalten, die Online-Frachtenbörse in einem kostenlosen Echtbetrieb zu testen.
- Die Online-Frachtenbörse muss die Abgabe und den Erhalt von Angeboten in Echtzeit ermöglichen.
- Die Online-Frachtenbörse muss einen Reklamations- und Inkassoservice anbieten.
- Die Online-Frachtenbörse muss die Sicherheit bei der Nutzung der Online-Frachtenbörse gewährleisten.
- Die Online-Frachtenbörse muss den Nutzern zusätzliche Leistungen anbieten, wie z.B. den Direktversand von Faxen und SMS.
- Die Online-Frachtenbörse muss den Nutzern ermöglichen, ihren Partnern Angebote zukommen zu lassen. Dies muss auch gelten, wenn diese Partner selbst die Online-Frachtenbörse nicht nutzen.
- Die Online-Frachtenbörse muss den Nutzern das Erstellen interner geschlossener Benutzergruppen innerhalb der Online-Frachtenbörse ermöglichen.

Darüber hinaus identifizieren die Autoren weitere Kriterien anhand von Experteninterviews.²⁸⁶

Aufgrund des Fehlens einer Online-Frachtenbörse für den Schienengüterverkehr – und damit des Fehlens von Wettbewerbern – zum Zeitpunkt ihrer Studie konnten HABIB und BRUNS im Rahmen ihrer Arbeit keine Umfeldanalyse des Wettbewerbs durchführen.²⁸⁷ Die stattdessen erfolgte Stakeholderanalyse ergab Anforderungen der potentiellen Stakeholder an eine Online-Frachtenbörse für den Schienengüterverkehr, die hier verkürzt wiedergegeben werden.²⁸⁸

Die Autoren identifizieren Anforderungen in verschiedenen Bereichen.²⁸⁹ Die befragten potentiellen Stakeholder haben die wichtigsten Anforderungen abschließend in eine Rangfolge gebracht.²⁹⁰

²⁸⁶ Vgl. HABIB/BRUNS (2012), S. 24 ff.

²⁸⁷ Vgl. HABIB/BRUNS (2012), S. 17.

²⁸⁸ Vgl. HABIB/BRUNS (2012), S. 17.

²⁸⁹ HABIB und BRUNS identifizieren Anforderungen in den Bereichen Technologie, Aufbau, Finanzierung, Leistung, Zusatzleistung, Sicherheit sowie Personal. Vgl. HABIB/BRUNS (2012), S. 20 f.

²⁹⁰ Vgl. HABIB/BRUNS (2012), S. 48.

Diese Anforderungen sind (absteigend nach Wichtigkeit sortiert):

1. Gewährleistung der Datensicherheit,
2. Aktualität der Angebote,
3. einfache Bedienbarkeit,
4. detaillierte Suchfunktion,
5. Internationalität,
6. jederzeitige Erreichbarkeit der Webseite,
7. Aufzeigen von Preisvergleichen,
8. Möglichkeit zur Kontaktknüpfung zu potentiellen Geschäftspartnern sowie
9. ansprechendes Design der Webseite.

An der von HABIB und BRUNS durchgeführten Stakeholderanalyse haben keine Frachtführer teilgenommen.²⁹¹

Die befragten potentiellen Stakeholder wurden weiterhin zu Vor- und Nachteilen einer Online-Frachtenbörse für den Schienengüterverkehr befragt.

Als Vorteile gaben sie an, dass eine Online-Frachtenbörse bei der Suche nach neuen Kooperationspartnern und beim Abbau der im Schienengüterverkehrsmarkt vorherrschenden Informationsasymmetrien und -defizite helfen könne.²⁹² Sie könne ebenfalls bei der Überwindung von Sprachbarrieren und Distanzen helfen sowie die Vermarktung freier Kapazitäten, Organisation von Rückladungen und Realisierung neuer Zugsysteme unterstützen.²⁹³ Zudem würde durch die Etablierung ein Wettbewerbsnachteil gegenüber dem Straßengüterverkehr eliminiert.²⁹⁴

Als Nachteile gaben sie an, dass die Gefahr besteht, dass Nutzer Aufträge an Wettbewerber verlieren könnten, denn durch die Nutzung der Online-Frachtenbörse könnten sich potentielle Kunden direkt an potentielle Dienstleister wenden.²⁹⁵ Der Schutz des eigenen Kundenstamms ist durch Nutzung einer Online-Frachtenbörse aus Sicht der potentiellen Nutzer gefährdet. Weiterhin wurde die Befürchtung geäußert, dass eine Online-Frachtenbörse dem Preisdumping im Schienengüterverkehr Vorschub leisten könnte.²⁹⁶

²⁹¹ Vgl. HABIB/BRUNS (2012), S. 48.

²⁹² Vgl. HABIB/BRUNS (2012), S. 57.

²⁹³ Vgl. HABIB/BRUNS (2012), S. 58.

²⁹⁴ Vgl. HABIB/BRUNS (2012), S. 58.

²⁹⁵ Vgl. HABIB/BRUNS (2012), S. 59.

²⁹⁶ Vgl. HABIB/BRUNS (2012), S. 59.

In ihrem Fazit halten HABIB und BRUNS fest, dass keiner der befragten potentiellen Stakeholder den Aspekt der Vermarktung einer Online-Frachtenbörse angesprochen hat.²⁹⁷

2.4.4 Branchenanalyse

Da die jüngsten Versuche, dedizierte Online-Frachtenbörsen für den Schienengüterverkehr zu etablieren, erst wenige Jahre alt sind, gestaltet sich die Analyse dieser Branche als schwierig. Noch im Jahr 2013 attestieren FIDAN und KUHLMANN, dass eine derartige Branche nicht existiert.²⁹⁸ Dennoch identifizieren die Autoren bereits Eintrittsbarrieren für die zu erwartenden Marktteilnehmer.²⁹⁹

Etwa 83% der Marktplatzbetreiber von Online-Frachtenbörsen sind als neutrale Betreiber zu betrachten.³⁰⁰ Sie nehmen als Marktplatzbetreiber die Aufgabe eines Intermediärs wahr, der eine unterstützende Funktion bei der Anbahnung von Verträgen ausübt, jedoch selbst keinen Teil der vermittelten Transportdienstleistung erbringt und nicht Teil der resultierenden Liefer- und Absatzkette ist.³⁰¹

Die Autoren geben zu bedenken, dass das Erreichen kostendeckender Erträge für Internetunternehmen zwar länger dauert, diese aber überproportional steigende Gewinne erzielen können, wenn sie sich am Markt behaupten.³⁰²

Der noch nicht erschlossene Markt für Online-Frachtenbörsen für den Schienengüterverkehr weist noch keine Produktdifferenzierung auf.³⁰³ Die existierenden Online-Frachtenbörsen könnten bereits über einen loyalen Kundenstamm verfügen, wodurch eine nicht näher untersuchte Kundenbindung an diese Unternehmen bestehen könnte.³⁰⁴ Dies würde relevant, falls diese Unternehmen sich in Zukunft dazu entscheiden würden, ihr bisheriges Angebot um den Bereich des Schienengüterverkehrs zu erweitern und so in Konkurrenz zu Online-Frachtenbörsen für den Schienengüterverkehr zu treten.³⁰⁵ Die Autoren weisen ebenfalls darauf hin, dass die Loyalität der Kunden im Falle eines auf Informationshandel spezialisierten Internetunternehmens nicht den selben Grad erreichen wird wie

²⁹⁷ Vgl. HABIB/BRUNS (2012), S. 60.

²⁹⁸ Vgl. FIDAN/KUHLMANN (2013), S. 8.

Zwei aktuelle Versuche der Etablierung von Online-Frachtenbörsen für den Schienengüterverkehr starteten erst nach der Veröffentlichung dieser Studie. Es wird dennoch für den Aspekt Branchenanalyse auf diese Studie zurückgegriffen, da jüngere Veröffentlichungen zu den gegründeten Online-Frachtenbörsen eher werbenden als wissenschaftlichen Charakter haben. Vgl. DÖRR/ENDEMANN (2014), S. 61 f.

²⁹⁹ Vgl. FIDAN/KUHLMANN (2013), S. 10 ff.

³⁰⁰ Vgl. FIDAN/KUHLMANN (2013), S. 7.

³⁰¹ Vgl. FIDAN/KUHLMANN (2013), S. 7.

³⁰² Vgl. FIDAN/KUHLMANN (2013), S. 11.

³⁰³ Vgl. FIDAN/KUHLMANN (2013), S. 11.

³⁰⁴ Vgl. FIDAN/KUHLMANN (2013), S. 11 f.

³⁰⁵ Vgl. FIDAN/KUHLMANN (2013), S. 12.

bei traditionellen Dienstleistungsunternehmen. Als Gründe hierfür werden die steigende Markttransparenz im Internet, der hierdurch steigende Informationsgrad der Kunden sowie sinkende Markteintrittsbarrieren und sich hieraus ergebende Wechsel- und Vergleichsmöglichkeiten angegeben.³⁰⁶

FIDAN und KUHLMANN führen weiter aus, dass ein Internetunternehmen einen geringeren Kapitalbedarf hat als ein Unternehmen, welches einen traditionellen Markteintritt plant.³⁰⁷ Es wird ausgeführt, dass es oftmals schnell und kostengünstig möglich sei, ein Internetunternehmen zu gründen, und dass das eingesetzte Kapital nicht in Lager oder physische Produktionsstätten investiert, sondern vornehmlich zur Deckung von Fixkosten aufgewendet werden muss.³⁰⁸

Zuzustimmen ist der Auffassung der Autoren, dass die Umstellungs- und Wechselkosten für Frachtenbörsen als gering anzusehen sind.³⁰⁹ Dieser Umstand fördert die als gering erachtete Zahlungsbereitschaft der potentiellen Kunden einer Online-Frachtenbörse und stellt ein potentiell Problem bei der Etablierung eines neuen Unternehmens dar.³¹⁰

Auch sind die Kunden von Internetunternehmen nicht in dem Maße auf den Zugang zu Vertriebskanälen angewiesen wie dies Kunden traditioneller Unternehmen sind. Anbieter und Kunden haben über das Internet die Möglichkeit direkt miteinander in Kontakt zu treten. Anbieter können so ihre Dienstleistung direkt an ihre potentiellen Kunden vermarkten. Kunden können sich ihrerseits selbstständig über verschiedene Angebote informieren und ohne physische Absatzmittler ihre Entscheidung für einen Anbieter treffen.

Für den Fall einer Online-Frachtenbörse bedeutet dies, dass Verlader, Frachtführer und Spediteure selbstständig eine Analyse der Konditionen im Markt der Frachtenbörsen durchführen können.³¹¹

Die Konzentration der Wettbewerber wird als hoch eingeschätzt, da zwar die Zahl der in Europa agierenden Online-Frachtenbörsen als im höheren zweistelligen Bereich angegeben wird, die Branche jedoch von wenigen Unternehmen dominiert wird, welche den Großteil des Marktes auf sich vereinen.³¹² Die hohe Konzentration innerhalb der Branche wird als Zeichen dafür gewertet, dass ein neu am Markt agierendes Unternehmen es schwer haben wird, in den Markt einzutreten.³¹³

³⁰⁶ Vgl. FIDAN/KUHLMANN (2013), S. 12.

³⁰⁷ Vgl. FIDAN/KUHLMANN (2013), S. 12.

³⁰⁸ Vgl. FIDAN/KUHLMANN (2013), S. 12.

³⁰⁹ Vgl. FIDAN/KUHLMANN (2013), S. 12 f.

³¹⁰ Vgl. FIDAN/KUHLMANN (2013), S. 12 f.

³¹¹ Vgl. FIDAN/KUHLMANN (2013), S. 13.

³¹² Vgl. FIDAN/KUHLMANN (2013), S. 16.

³¹³ Vgl. FIDAN/KUHLMANN (2013), S. 16.

Die geringe Attraktivität des Schienengüterverkehrs könnte als Grund für das Fehlen einer etablierten Branche für Online-Frachtenbörsen für den Schienengüterverkehr gesehen werden.³¹⁴ Die starke Dynamik innerhalb der Branche der Online-Frachtenbörsen lässt vermuten, dass eine Branche für Online-Frachtenbörsen für den Schienengüterverkehr, sollte sie sich etablieren, ebenfalls eine hohe Dynamik aufweisen wird.³¹⁵

Eine Chance kann in der Differenzierung einer Online-Frachtenbörse für den kombinierten Verkehr gesehen werden, da eine solche Online-Frachtenbörse ihren Kunden durch die größere Wahlfreiheit bzgl. möglicher Vertragsabschlüsse und Transportwege eine differenziertere Entscheidung ermöglichen und sich so vom Angebot der bestehenden Unternehmen abgrenzen würde.³¹⁶

Für eine Online-Frachtenbörse können keine generell relevanten Marktaustrittsbarrieren identifiziert werden.³¹⁷

Nach einem erfolgreichen Markteintritt wird der Druck durch Substitutionsprodukte als Gefahr für den Geschäftserfolg identifiziert.³¹⁸ Für eine Online-Frachtenbörse für den kombinierten Verkehr könnten in Zukunft andere Online-Frachtenbörsen, bspw. für die Binnenschifffahrt, solche Substitutionsprodukte darstellen.³¹⁹ Des Weiteren könnten Kunden bei Unzufriedenheit wieder zu den zuvor von ihnen genutzten Online-Frachtenbörsen zurückkehren.³²⁰ Die Online-Frachtenbörse für den kombinierten Verkehr könnte jedoch auch selbst ein Substitutionsprodukt für Online-Frachtenbörsen für den Straßengüterverkehr sein, wenn sie es schafft, die Vermittlung einer weiteren Transportart als Differenzierungskriterium gegenüber diesen Online-Frachtenbörsen zu vermarkten.³²¹

Der Druck durch Nachahmer ist laut FIDAN und KUHLMANN besonders hervorzuheben, da Unternehmen, die dem E-Business zuzuordnen sind, sich besonders häufig mit Nachahmungen ihrer Geschäftsideen konfrontiert sehen.³²² Die Autoren vermuten, dass der Quelltext einer Webseite potentiellen Konkurrenten detaillierte Informationen über den Aufbau der Webseite preisgibt, was wiederum zu einem schnellen und einfachen Kopieren der Geschäftsidee oder auch der Webseite führen könnte.³²³

³¹⁴ Vgl. FIDAN/KUHLMANN (2013), S. 16.

³¹⁵ Vgl. FIDAN/KUHLMANN (2013), S. 16 f.

³¹⁶ Vgl. FIDAN/KUHLMANN (2013), S. 17.

³¹⁷ Vgl. FIDAN/KUHLMANN (2013), S. 18.

³¹⁸ FIDAN und KUHLMANN führen aus, dass die Produkte von Internetunternehmen oft in Konkurrenz zu anderen Produkten stehen, die füreinander Substitute darstellen. Vgl. FIDAN/KUHLMANN (2013), S. 19.

³¹⁹ Vgl. FIDAN/KUHLMANN (2013), S. 19.

³²⁰ Vgl. FIDAN/KUHLMANN (2013), S. 19.

³²¹ Vgl. FIDAN/KUHLMANN (2013), S. 20.

³²² Vgl. FIDAN/KUHLMANN (2013), S. 20.

³²³ Vgl. FIDAN/KUHLMANN (2013), S. 20.

Dieser Ansicht muss in Teilen widersprochen werden: Zwar sehen sich insbesondere populäre Internetunternehmen häufig Nachahmern ihres Geschäftskonzeptes ausgesetzt, jedoch eignen sich die durch den Quelltext der Webseite preisgegebenen Informationen in den seltensten Fällen zur Nachahmung einer Geschäftsidee. Grund hierfür ist die Tatsache, dass die entscheidenden Strukturen und Prozesse eines Internetunternehmens in der Regel nicht auf der Webseite selbst, sondern auf den Servern des Unternehmens stattfinden, wie bspw. spezielle Auktionsalgorithmen oder das Matching von Angebot und Nachfrage im Rahmen eines Marktplatzes. Solange ein potentieller Nachahmer sich nicht mit illegalen Mitteln Zugriff auf diesen Teil des Quelltexts eines Unternehmens verschafft, sind die ihm durch die reguläre Webseite preisgegebenen Informationen als trivial zu erachten. Der Aussage, dass mit Hilfe der durch den Quelltext der Webseite preisgegebenen Informationen die Optik und Teile der Benutzerführung der Webseite selbst relativ einfach kopiert werden kann, muss jedoch zugestimmt werden.

Die Macht der Nutzer ist als sehr groß einzuschätzen, da durch die stattfindende Disintermediation die Verhandlungsmacht der Nutzer aufgrund des Fehlens physischer Absatzmittler und der gestiegenen Markttransparenz gestärkt wird.³²⁴ Auch die geringen Umstellungs- und Wechselkosten deuten auf diesen Umstand hin.³²⁵

Erschwerend kommt hinzu, dass die Nutzer der Online-Frachtenbörse gleichzeitig als Anbieter und Nachfrager auftreten.³²⁶ Die Betreiber sehen sich somit sowohl mächtigen Abnehmern als auch mächtigen Lieferanten von Informationen gegenüber, da diese Gruppen sich im Falle einer Online-Frachtenbörse nicht strikt voneinander trennen lassen.³²⁷

Positiv zu bewerten ist, dass das Thema „kombinierter Verkehr“ in Deutschland politische Relevanz besitzt.³²⁸

Negativ dagegen erscheint, dass der kombinierte Verkehr keine stetig ansteigende Tendenz erkennen lässt und der Verkehrsträger Straße der mit Abstand meistgenutzte Güterverkehrsträger in Deutschland ist, während der Schienengüterverkehr einen geringen Anteil am gesamten Transportaufkommen hat.³²⁹

Das Geschäftsmodell der Online-Frachtenbörsen wird von den potentiellen Nutzern akzeptiert.³³⁰ Jedoch wäre für eine Online-Frachtenbörse für den kombinierten Verkehr nicht nur die Akzeptanz von Online-Frachtenbörsen, sondern auch die des Schienengüterverkehrs

³²⁴ Vgl. FIDAN/KUHLMANN (2013), S. 20 f.

³²⁵ Vgl. FIDAN/KUHLMANN (2013), S. 21.

³²⁶ Vgl. FIDAN/KUHLMANN (2013), S. 21.

³²⁷ Vgl. FIDAN/KUHLMANN (2013), S. 21.

³²⁸ Vgl. FIDAN/KUHLMANN (2013), S. 24.

³²⁹ Vgl. FIDAN/KUHLMANN (2013), S. 24 f.

³³⁰ Vgl. FIDAN/KUHLMANN (2013), S. 32.

entscheidend.³³¹ Sofern der Schienengüterverkehr als mögliches Transportmittel nicht attraktiv erscheint, ist die Nutzung einer Online-Frachtenbörse für den kombinierten Verkehr nicht zu erwarten.³³² Die von der Politik vorgebrachten Umweltaspekte einer Verlagerung des Transports von der Straße auf die Schiene dürften für die potentiellen Teilnehmer wenig Relevanz besitzen, wenn die wirtschaftlichen Aspekte eines Transports über die Schiene mit Kostennachteilen verbunden sind.³³³

Bei der Nutzung von Online-Frachtenbörsen ist darauf zu achten, dass keine unnötigen Sprachbarrieren bei der Nutzung der Webseite entstehen.³³⁴ FIDAN und KUHLMANN identifizieren Indikatoren dafür, dass mögliche Sprachbarrieren die Geschäftstätigkeit einer Online-Frachtenbörse beeinträchtigen könnte, da die Bereitschaft der Nutzer, Geschäftsabschlüsse in einer Fremdsprache vorzunehmen, als gering eingeschätzt wird.³³⁵

Eine Online-Frachtenbörse muss sich mit den Problemen der Datensicherheit und der Sicherheit der Geschäftsabschlüsse auseinandersetzen.³³⁶ Das Vertrauen der Nutzer in die Integrität ihrer Daten sowie die Verlässlichkeit und Vertraulichkeit der über die Online-Frachtenbörse gefassten Geschäftsabschlüsse stellen entscheidende Kriterien für den Geschäftserfolg einer Online-Frachtenbörse dar und müssen dem potentiellen Nutzer glaubhaft – bspw. mit Hilfe von Zertifizierungen und eines ausführlichen Screenings der Teilnehmer – vermittelt werden.³³⁷

FIDAN und KUHLMANN empfehlen aufgrund dieser Erkenntnisse keinen der von Porter³³⁸ empfohlenen generischen Strategietypen, sondern sehen in der Verfolgung einer hybriden Strategie die beste Wahl.³³⁹ Begründet wird dies mit der mangelnden Zahlungsbereitschaft der Kunden und der mangelnden Produktdifferenzierung im Bereich der Online-Frachtenbörsen.³⁴⁰ So wird empfohlen, den Nutzern ein differenziertes Produkt, welches die Nutzung von Straßen- und Schienengüterverkehr ermöglicht, kostenfrei zur Verfügung zu stellen und die zum Geschäftsbetrieb notwendigen Einnahmen aus Werbung zu generieren.³⁴¹

Diese Empfehlung ist kritisch zu hinterfragen, da – wie bereits aufgezeigt – die Kosten der Etablierung und des Betriebs einer ausreichend differenzierten Online-Frachtenbörse

³³¹ Vgl. FIDAN/KUHLMANN (2013), S. 32.

³³² Vgl. FIDAN/KUHLMANN (2013), S. 32.

³³³ Vgl. FIDAN/KUHLMANN (2013), S. 32.

³³⁴ Vgl. FIDAN/KUHLMANN (2013), S. 32.

³³⁵ Vgl. FIDAN/KUHLMANN (2013), S. 33.

³³⁶ Vgl. FIDAN/KUHLMANN (2013), S. 33 f.

³³⁷ Vgl. FIDAN/KUHLMANN (2013), S. 33 f.

³³⁸ Vgl. PORTER (2008).

³³⁹ Vgl. FIDAN/KUHLMANN (2013), S. 40 f.

³⁴⁰ Vgl. FIDAN/KUHLMANN (2013), S. 40 f.

³⁴¹ Vgl. FIDAN/KUHLMANN (2013), S. 41.

nicht unterschätzt werden sollten. Es erscheint zumindest fraglich, ob die notwendigen Einnahmen durch das Einblenden von Werbung realisiert werden können. Grund hierfür ist die als gering eingeschätzte Zahlungsbereitschaft der Nutzer und die Tatsache, dass diese Nutzer aus Sicht der potentiell Werbenden kein lukratives Zielpublikum darstellen werden.

FIDAN und KUHLMANN stellen fest, dass die von ihnen analysierte Branchenstruktur für Online-Frachtenbörsen für den kombinierten Verkehr nur eine mögliche Variante darstellt, da die Branche noch nicht existiert und bereits minimale Veränderungen der Umweltbedingungen sowohl die getroffenen Annahmen als auch das Ergebnis der Analyse beeinflussen könnten.³⁴² Als Beispiele für derartige Veränderungen werden vor allem technische und rechtliche Entwicklungen im Bereich des Straßengüterverkehrs angeführt. So könnte die Attraktivität dieser Transportart weiter steigen, wenn sich in Zukunft restriktivere Normen bzgl. des Schadstoffausstoßes durchsetzen oder die Verwendung längerer Lastkraftwagen – sog. Gigaliner – legalisiert wird.³⁴³ Die hieraus resultierenden positiven Effekte für den Straßengüterverkehr könnten sich negativ auf den Schienengüterverkehr und den kombinierten Verkehr auswirken.³⁴⁴

Abschließend stellen FIDAN und KUHLMANN fest, dass es für eine Online-Frachtenbörse für den kombinierten Verkehr durchaus möglich sei, sich auf dem Markt zu etablieren, die Betreiber jedoch mit Schwierigkeiten rechnen müssten, wenn sie hohe Marktanteile gewinnen wollen.³⁴⁵

2.4.5 Abschätzung des Marktpotentials

Die Abschätzung des Marktpotentials einer Online-Frachtenbörse für Schienengüterverkehre auf der Strecke zwischen Rotterdam und Genua ist ein wichtiger Teil der Evaluierung des potentiellen Marktplatzes. Sie gibt Aufschluss darüber, welche Kennzahlen in Bezug auf Marktgröße, Teilnehmerzahl und Umsatz zu erwarten sind.³⁴⁶

BRUNS et al. betrachten die Güterbeförderungsleistungen in den betreffenden Ländern – Niederlande, Deutschland, Schweiz sowie Italien – und vergleichen die Güterbeförderungsleistungen in den Jahren 2000 bis 2010.³⁴⁷ Sie stellen fest, dass die Güterbeförderungsleistungen durch die im Jahr 2008 begonnene Wirtschaftskrise im Jahr 2009 einen deutlichen Einbruch erlitten haben.³⁴⁸ Ein im Jahr 2010 zu verzeichnender leichter An-

³⁴² Vgl. FIDAN/KUHLMANN (2013), S. 42.

³⁴³ Vgl. FIDAN/KUHLMANN (2013), S. 42 f.

³⁴⁴ Vgl. FIDAN/KUHLMANN (2013), S. 42.

³⁴⁵ Vgl. FIDAN/KUHLMANN (2013), S. 42.

³⁴⁶ Vgl. BRUNS et al. (2012a), S. 8 f.

³⁴⁷ Vgl. BRUNS et al. (2012a), S. 2.

³⁴⁸ Vgl. BRUNS et al. (2012a), S. 2.

stieg konnte die Einbußen des Vorjahres nicht ausgleichen.³⁴⁹ Die angeführten Prognosen sehen einen Anstieg der Güterbeförderungsleistung sowohl in Deutschland als auch in Europa.³⁵⁰

Tabelle 1 zeigt Prognosen zur Güterbeförderungsleistung in Deutschland.

	2005	2020	2030
Straßengüterverkehr	404	605	707
Schienengüterverkehr	95	143	170
Binnenschifffahrt	64	78	87
Rohrfernleitungen	17	17	17
Gesamtleistung	581	842	982

Tabelle 1: Prognosen zur Güterbeförderungsleistung in Deutschland (in Mrd. Tonnenkilometern)³⁵¹

Für den langfristigen Erfolg einer Online-Frachtenbörse ist eine ausreichende Teilnehmerzahl, deren Zusammensetzung aus den verschiedenen Akteursgruppen sowie ihre Entwicklung entscheidend.³⁵² Die beteiligten Akteursgruppen im Schienengüterverkehr sind – wie bereits beschrieben³⁵³ – Eisenbahnverkehrsunternehmen, produzierende Unternehmen (Verlader), Frachtführer und Spediteure.³⁵⁴ Die Aufgaben dieser Gruppen überschneiden sich teilweise, da Verlader und Spediteure ebenfalls im Straßengüterverkehr tätig sind.³⁵⁵

Über die Branchenstrukturen dieser Akteursgruppen lassen sich nur ungenaue Angaben machen. So gibt es in Deutschland ca. 300 Eisenbahnverkehrsunternehmen, von denen etwa 100 Güterverkehre abwickeln.³⁵⁶ Im Jahr 2009 gab es in Deutschland ca. 1.400.000 Unternehmen in Wirtschaftszweigen, die vorrangig auf der Schiene vertreten sind (potentielle Verlader).³⁵⁷ Darüber hinaus existieren ca. 15.000 Speditionen, von denen etwa 2.000 Bahnen befrachten.³⁵⁸

³⁴⁹ Vgl. BRUNS et al. (2012a), S. 2.

³⁵⁰ Hierbei ist zu beachten, dass die angeführten, bis ins Jahr 2025 reichenden Prognosen aus dem Jahr 2007 stammen. Sie entstanden somit vor der Wirtschaftskrise im Jahr 2008. Vgl. BRUNS et al. (2012a), S. 3.

³⁵¹ Quelle: eigene Darstellung in Anlehnung an ICKERT et al. (2007), S. 118.

³⁵² Vgl. BRUNS et al. (2012a), S. 5.

³⁵³ Vgl. Kapitel 2.1, S. 7.

³⁵⁴ Vgl. BRUNS et al. (2012a), S. 5.

³⁵⁵ Vgl. BRUNS et al. (2012a), S. 5.

³⁵⁶ Vgl. BRUNS et al. (2012a), S. 6.

³⁵⁷ Vgl. BRUNS et al. (2012a), S. 6.

³⁵⁸ Vgl. BRUNS et al. (2012a), S. 6.

Da Online-Frachtenbörsen im Schienengüterverkehr noch nicht verbreitet sind, werden im Rahmen der Analyse lediglich Online-Frachtenbörsen im Straßengüterverkehr betrachtet.³⁵⁹ Das Leistungsangebot dieser „Lkw-Online-Frachtenbörsen“ ist sehr heterogen.³⁶⁰ So existieren Plattformen, welche Ausschreibungen im Stil eines „schwarzen Bretts“ ermöglichen, sowie Plattformen, welche Auktionen durchführen, und solche, welche beide Funktionen anbieten.³⁶¹

BRUNS et al. identifizieren zwei bei Lkw-Online-Frachtenbörsen anzutreffende Geschäfts- und Preismodelle: das transportvolumen- und transaktionsabhängige Modell sowie das nutzer- und zeitabhängige Modell.³⁶²

Tabelle 2 zeigt ausgewählte Lkw-Online-Frachtenbörsen und ihr Leistungsangebot.

Name	Leistung	transportvolumen- und transaktionsabhängige Abrechnung	nutzer- und zeitabhängige Abrechnung
www.benelog.de	Ausschreibungen, Auktionen	3% des Auftragsvolumens für Frachtführer	monatliche Grundgebühr für Frachtführer und Verloader
www.cargoclix.de	Ausschreibungen, Auktionen	Stufenpreise je Ladung/Palette	
www.nolis.com	Auktionen	Volumenpreise direkt zwischen Vertragspartnern verhandelbar	je Monat und Tool für Benutzung der Plattform
www.teleroute.de	Auktionen		monatlicher Festpreis
www.timocom.de	Ausschreibungen, Auktionen		monatlicher Festpreis

Tabelle 2: Preis- und Geschäftsmodelle ausgewählter Lkw-Online-Frachtenbörsen³⁶³

Zur weiteren Informationsgewinnung wurde eine Befragung von sechs entscheidungsbefugten Branchenexperten durchgeführt.³⁶⁴ Analog zu vorherigen Analysen wurde zunächst der Markt für straßengebundene Transporte betrachtet und anschließend versucht, Rückschlüsse auf die Potentiale des Schienengüterverkehrs zu ziehen.

Die Steigerung der Güterbeförderungsleistung von 62 Prozent im Straßengüterverkehr halten die Experten für zu hoch, realistischer erscheint ihnen ein Wachstum im Bereich

³⁵⁹ Vgl. BRUNS et al. (2012a), S. 7.

³⁶⁰ Vgl. BRUNS et al. (2012a), S. 7.

³⁶¹ Vgl. BRUNS et al. (2012a), S. 7.

³⁶² Vgl. BRUNS et al. (2012a), S. 7.

³⁶³ Quelle: eigene Darstellung in Anlehnung an BRUNS et al. (2012a), S. 7.

³⁶⁴ Vgl. BRUNS et al. (2012a), S. 11.

von 10 bis 50 Prozent.³⁶⁵ Den Anteil von Online-Frachtenbörsen am Transportaufkommen im Straßengüterverkehr schätzen die Experten auf 5 bis 20 Prozent.³⁶⁶ Ferner handele es sich bei den gehandelten Gütern um kurzfristig zu transportierende Kleinstmengen sowie die Handhabung von Spitzenauslastungen.³⁶⁷

Die Experten schätzen den Anteil der Marktteilnehmer, die derzeit eine Online-Frachtenbörse für den Straßengüterverkehr nutzen, als hoch ein.³⁶⁸ Grund hierfür seien die dezentralen Strukturen der kleinen und mittelgroßen Speditionen und Frachtführer, für die das Internet ein geeignetes Medium zur Geschäftsabwicklung darstelle.³⁶⁹

Im Gegensatz zum Schienengüterverkehr sind die produzierenden Unternehmen im Straßengüterverkehr kaum in die Transportabwicklung involviert, Speditionen und Frachtführer verhandeln meist direkt miteinander.³⁷⁰ BRUNS et al. fassen zusammen, dass es im Bereich der Lkw-Online-Frachtenbörsen viele Nutzer gibt, welche die Frachtenbörsen lediglich zur Abwicklung von Kleinstmengen sowie zur Informationsgewinnung einsetzen.

Die Steigerung der Güterbeförderungsleistung von 42 Prozent im Schienengüterverkehr halten die Experten – wie schon die Prognose für den Straßengüterverkehr – für zu hoch, auch ein Rückgang des Straßengüterverkehrs zugunsten des Schienengüterverkehrs erscheint ihnen unwahrscheinlich.³⁷¹ Realistischer erscheint ihnen ein Wachstum im Bereich von 5 bis 30 Prozent bei konstant bleibendem Modal Split.³⁷²

Die Experten merken an, dass es in der Vergangenheit einige staatlich geförderte Versuche zur Etablierung einer Online-Frachtenbörse für den Schienengüterverkehr gab. Sie rechnen damit, dass es in Zukunft mehrere Online-Frachtenbörsen für den Schienengüterverkehr geben wird, deren Etablierung jedoch abhängig vom Erreichen der kritischen Masse an Nutzern sowie der Identifizierung des „richtigen“ Geschäftsmodells sei.

Es erscheint den befragten Experten fraglich, ob Online-Frachtenbörsen für den Schienengüterverkehr unabhängig von Online-Frachtenbörsen für den Straßengüterverkehr funktionieren können.³⁷³ Sie geben zu bedenken, dass die Betreiber etablierter Lkw-Online-Frachtenbörsen ihr Leistungsangebot relativ einfach erweitern könnten, um so mit der „Frachtenbörse von morgen“ beide Verkehrsträger abzudecken.³⁷⁴ In jedem Fall werde die

³⁶⁵ Vgl. BRUNS et al. (2012a), S. 11.

³⁶⁶ Vgl. BRUNS et al. (2012a), S. 11.

³⁶⁷ Vgl. BRUNS et al. (2012a), S. 11.

³⁶⁸ Es werden Zahlen zwischen 30 und 100 Prozent genannt. Vgl. BRUNS et al. (2012a), S. 12.

³⁶⁹ Vgl. BRUNS et al. (2012a), S. 12.

³⁷⁰ Vgl. BRUNS et al. (2012a), S. 12.

³⁷¹ Vgl. BRUNS et al. (2012a), S. 12.

³⁷² Vgl. BRUNS et al. (2012a), S. 12.

³⁷³ Vgl. BRUNS et al. (2012a), S. 12.

³⁷⁴ Vgl. BRUNS et al. (2012a), S. 12.

Entwicklung einer Online-Frachtenbörse für den Schienengüterverkehr von Deutschland ausgehen.³⁷⁵

Die Experten prognostizierten weiter, dass das über Online-Frachtenbörsen für den Schienengüterverkehr abgewinkelte Transportvolumen geringer ausfallen wird als bei Lkw-Online-Frachtenbörsen.³⁷⁶ Als Gründe werden die höhere Auslastung der Wagen sowie die mangelnde Flexibilität in der Streckenführung genannt.³⁷⁷

Für den Schienengüterverkehrsmarkt erwarten die Experten zudem eine Zunahme des Wettbewerbs, welcher in Konzentrationsprozessen münden wird. Zur Etablierung internationaler Eisenbahnverkehrsunternehmen äußern die Experten keine einheitliche Prognose.³⁷⁸

Die Tatsache, dass aufgrund der direkten Verhandlungen zwischen produzierenden Unternehmen (Verladern) und Eisenbahnverkehrsunternehmen im Schienengüterverkehr weniger Speditionen eingeschaltet werden als im Straßengüterverkehr, halten die Experten für bedenklich.³⁷⁹ Hier sei eine Trendwende notwendig, Spediteure müssten in Zukunft vermehrt eine Befrachtung auch von Bahnen anbieten.³⁸⁰ Bei vermehrtem Wettbewerb sei eine direkte Befrachtung durch die Eisenbahnverkehrsunternehmen nicht mehr möglich, da der persönliche Kontakt zwischen produzierenden Unternehmen und Eisenbahnverkehrsunternehmen verloren ginge.³⁸¹ Diese Lücke könnte durch die Vermittlerrolle der Speditionen ersetzt werden.³⁸²

Die damit einhergehende Erweiterung des Kreises potentieller Benutzer sei zudem eine notwendige Voraussetzung für das Funktionieren einer Online-Frachtenbörse im Schienengüterverkehr.³⁸³

Auf die Frage, welches Geschäfts- und Preismodell sie für den Betrieb einer Online-Frachtenbörse im Schienengüterverkehr empfehlen würden, antworteten die Experten, dass sie nur das ausschreibungsbasierte Geschäftsmodell für im Schienengüterverkehr tragfähig erachten.³⁸⁴ Da Eisenbahnen – im Vergleich zu Lkw – nicht so flexibel und kurzfristig reagieren könnten, seien Plattformen für Schienengüterverkehre eher für den Einsatz bei Linienverkehren sowie kombinierten Verkehren geeignet.³⁸⁵ Ein nutzer- und zeit-

³⁷⁵ Vgl. BRUNS et al. (2012a), S. 12.

³⁷⁶ Vgl. BRUNS et al. (2012a), S. 13.

³⁷⁷ Vgl. BRUNS et al. (2012a), S. 13.

³⁷⁸ Vgl. BRUNS et al. (2012a), S. 13.

³⁷⁹ Vgl. BRUNS et al. (2012a), S. 13.

³⁸⁰ Vgl. BRUNS et al. (2012a), S. 13.

³⁸¹ Vgl. BRUNS et al. (2012a), S. 13.

³⁸² Vgl. BRUNS et al. (2012a), S. 13.

³⁸³ Vgl. BRUNS et al. (2012a), S. 13.

³⁸⁴ Vgl. BRUNS et al. (2012a), S. 14.

³⁸⁵ Vgl. BRUNS et al. (2012a), S. 14.

abhängiges Preismodell in Form einer „Flatrate“ erscheint den Experten besser als ein transportvolumen- und transaktionsabhängiges, da es die Nutzer zu mehr Transaktionen bewegen würde.³⁸⁶

Der kombinierte Verkehr wird in Europa zukünftig eine größere Rolle spielen.³⁸⁷ Hierdurch würde eine Online-Frachtenbörse gefragt, welche mehrere Verkehrsträger – bspw. Straße und Schiene – bedient.³⁸⁸ Ob diese multimodale Online-Frachtenbörse aus den bestehenden Lkw-Online-Frachtenbörsen hervorgehen oder als Neuentwicklung auftreten wird, scheint ungewiss. Die Tatsache, dass keine bestehende Frachtenbörse ihr Angebot auf die Schiene erweitert hat, deutet daraufhin, dass den Betreibern der bestehenden Lkw-Online-Frachtenbörsen diese Erweiterung ihres Leistungsangebots unattraktiv erscheint. Der Grund hierfür ist, dass hierzu befragte Experten unterstellen, dass die bestehenden Betreiber ihre Frachtenbörsen leicht um ein Angebot für den Schienengüterverkehr erweitern könnten.³⁸⁹ In einem als „average case“ bezeichneten Szenario³⁹⁰ wird vorgeschlagen, dass die Entwickler einer neuen Online-Frachtenbörse für den Schienengüterverkehr eine Kooperation mit einer bestehenden Lkw-Online-Frachtenbörse eingehen und einen gemeinsamen Leistungskatalog bei getrennter Geschäftsführung anbieten sollten.³⁹¹ Alternativ könnten sie einen entwickelten Prototyp an den Betreiber einer Lkw-Online-Frachtenbörse verkaufen.³⁹²

Aus der von BRUNS et al. vorgenommenen Analyse des Marktpotentials für Online-Frachtenbörsen im Segment von Schienengüterverkehren und kombinierten Verkehren lässt sich schlussfolgern, dass die Marktchancen einer Online-Frachtenbörse für den kombinierten Verkehr, welche mehrere Verkehrsträger abdeckt, höher einzuschätzen sind als jene einer Online-Frachtenbörse, welche nur den Schienengüterverkehr abdeckt.

³⁸⁶ Vgl. BRUNS et al. (2012a), S. 14.

³⁸⁷ Vgl. ICKERT et al. (2007), S. 61 f. u. 99.

³⁸⁸ Vgl. BRUNS et al. (2012a), S. 17.

³⁸⁹ Vgl. BRUNS et al. (2012a), S. 17.

³⁹⁰ BRUNS et al. beschreiben zwei weitere *cases*, welche hier nicht explizit betrachtet werden. Vgl. BRUNS et al. (2012a), S. 16 f. Die Ausführungen zu beiden Szenarien finden aber Berücksichtigung in der weiteren Analyse. Der geschilderte *worst case* erscheint an dieser Stelle weniger relevant als der *average case*, da es den potentiellen Betreibern einer Online-Frachtenbörse für den Schienengüterverkehr in diesem Szenario nicht gelingt, die kritische Masse an Nutzern zu erreichen. Ein Scheitern aus diesem Grund wird als allgemeines Risiko des Betriebes eines Online-Marktplatzes angesehen und nicht als geeignet erachtet, um die weitere Analyse primär auf dieses Szenario zu stützen. Der geschilderte *best case* erscheint an dieser Stelle wenig geeignet für die weitere Analyse, da er umfassende Änderungen der politischen und wirtschaftlichen Rahmenbedingungen zugrundelegt. Die Eintrittswahrscheinlichkeit dieser Änderungen wird als zu gering erachtet, um die weitere Analyse primär auf dieses Szenario zu stützen.

³⁹¹ Vgl. BRUNS et al. (2012a), S. 17.

³⁹² Vgl. BRUNS et al. (2012a), S. 17.

2.4.6 Vorliegende Implementierung eines Softwareprototyps

2.4.6.1 Ziele

Im Rahmen des EU-Projekts CODE24 wurde vom Institut für Produktion und Industrielles Informationsmanagement im Zeitraum von 2010 bis 2012 der erste Prototyp einer Online-Frachtenbörse für den Schienengüterverkehr implementiert.³⁹³

Dieser Prototyp basierte in seiner ersten Version, die im Jahr 2011 veröffentlicht wurde, auf den von KLIPPERT et al. erhobenen Anforderungen.³⁹⁴ In den Jahren 2011 und 2012 wurde die Implementierung auf Grundlage des Feedbacks von Branchenexperten sukzessive angepasst und erweitert.³⁹⁵

Das vorrangige Ziel des Softwareprototyps bestand in der Implementierung der ermittelten Anforderungen an eine Online-Frachtenbörse für den Schienengüterverkehr zur Unterstützung der weiteren Anforderungsanalyse im Bereich „Online-Frachtenbörsen im Schienengüterverkehr“.³⁹⁶

Hierzu sollte der Softwareprototyp bestehende interne und externe Geschäftsprozesse seiner Nutzer unterstützen.³⁹⁷ Die Schaffung neuer Prozesse zur Nutzung der Online-Frachtenbörse sollte vermieden werden.³⁹⁸ Dennoch sollte der Softwareprototyp die Nutzer in ihrer Benutzung der Online-Frachtenbörse führen, um einerseits ein gewünschtes Verhalten – bspw. das Ausfüllen sämtlicher Kontaktinformationen – zu fördern und andererseits im Rahmen der weiteren Anforderungsanalyse bestimmte Funktionalitäten auf die mit ihnen einhergehenden Verhaltensweisen der Nutzer testen zu können.³⁹⁹

Die innerhalb des Softwareprototyps abgebildeten Prozesse wurden bewusst schlank gehalten und modular aufgebaut, um – unterstützt durch eine gute Strukturierung – eine spätere Erweiterung der Funktionalitäten zu unterstützen.⁴⁰⁰

Die Benutzeroberfläche wurde übersichtlich gestaltet und mit dem Ziel konzipiert, die einfache Handhabung zu erleichtern und möglichst intuitiv zugänglich zu sein.⁴⁰¹ Dabei wurde – zusätzlich zu dem für angemeldete Nutzer zugänglichen Bereich – ein nur

³⁹³ Vgl. FÖHRING et al. (2012), sowie FÖHRING/BRUNS (2011).

³⁹⁴ Vgl. FÖHRING/BRUNS (2011), S. 2.

³⁹⁵ Vgl. FÖHRING et al. (2012).

³⁹⁶ Vgl. FÖHRING/BRUNS (2011), S. 8.

³⁹⁷ Vgl. FÖHRING/BRUNS (2011), S. 8.

³⁹⁸ Vgl. FÖHRING/BRUNS (2011), S. 8.

³⁹⁹ Vgl. FÖHRING/BRUNS (2011), S. 8.

⁴⁰⁰ Vgl. FÖHRING/BRUNS (2011), S. 8 f.

⁴⁰¹ Vgl. FÖHRING/BRUNS (2011), S. 9.

den Betreibern der Online-Frachtenbörse zugänglicher administrativer Bereich implementiert.⁴⁰²

Abbildung 2 zeigt die Benutzeroberfläche.

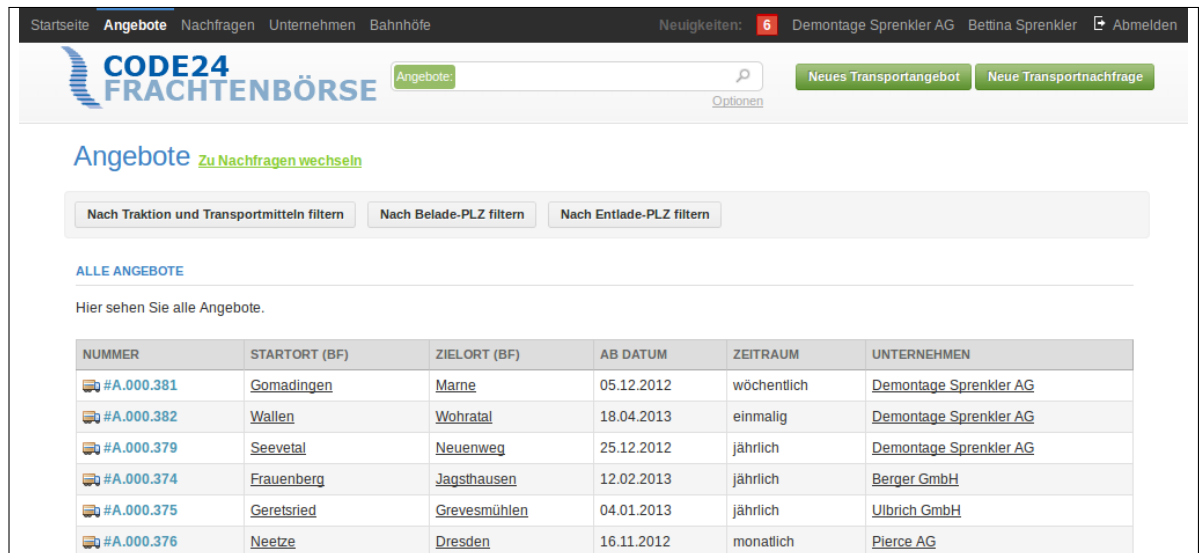


Abbildung 2: Benutzeroberfläche des Softwareprototyps aus dem EU-Projekt CODE24⁴⁰³

2.4.6.2 Leistungsumfang

Interessierte Nutzer können sich im System des Softwareprototyps registrieren.⁴⁰⁴ Alle im System des Softwareprototyps registrierten Unternehmen haben ein für alle anderen Nutzer einsehbares Unternehmensprofil.⁴⁰⁵

Eine Benutzerverwaltung ermöglicht den registrierten Unternehmen die Verwaltung mehrerer Benutzerkonten und der ihnen zugewiesenen Berechtigungen.⁴⁰⁶ Es wurde ein Rechtesystem implementiert, welches Benutzerrollen definiert und diese mit bestimmten Berechtigungen verknüpft, um den Zugriff auf Informationen und Funktionalitäten zu beschränken und betriebliche Funktionen im System des Softwareprototyps abzubilden.⁴⁰⁷ Das Rechtesystem des Softwareprototyps bestimmt die Zugriffsmöglichkeiten eines Benutzers auf Informationen innerhalb des Softwareprototyps sowie die Präsentation seiner individuellen Benutzeroberfläche.⁴⁰⁸

⁴⁰² Vgl. FÖHRING/BRUNS (2011), S. 9.

⁴⁰³ Quelle: eigene Darstellung.

⁴⁰⁴ Vgl. FÖHRING et al. (2012), S. 25, sowie FÖHRING/BRUNS (2011), S. 36.

⁴⁰⁵ Vgl. FÖHRING et al. (2012), S. 56.

⁴⁰⁶ Vgl. FÖHRING et al. (2012), S. 61 ff.

⁴⁰⁷ Vgl. FÖHRING/BRUNS (2011), S. 44.

⁴⁰⁸ Vgl. FÖHRING/BRUNS (2011), S. 45.

Die Nutzer des Softwareprototyps können zu ihrem Unternehmen sowie ihrer Person umfangreiche Kontaktinformationen im System hinterlegen.⁴⁰⁹ Der Softwareprototyp fördert die Hinterlegung dieser Informationen, da die Möglichkeit zum Einstellen möglichst vollständiger Kontaktinformationen als kritischer Erfolgsfaktor beim Betrieb einer Online-Frachtenbörse identifiziert wurde.⁴¹⁰

Registrierte Nutzer können Inserate für Transportangebote sowie Transportnachfragen erstellen.⁴¹¹ Die Inserate enthalten zum einen Meta-Informationen über das inserierende Unternehmen sowie über ihre Gültigkeitsdauer und Sichtbarkeit und zum anderen konkrete Angaben zur angebotenen oder nachgefragten Transportdienstleistung.⁴¹² Inserate können von interessierten Nutzern im System des Softwareprototyps gesichtet, gefiltert, beantwortet und verhandelt werden.⁴¹³ Hierbei sind für einen angemeldeten Nutzer nur jene Inserate sichtbar, bei denen der inserierende Nutzer explizit angegeben hat, dass der betreffende Nutzer zu einer Benutzergruppe gehört, welche das Inserat sehen darf.⁴¹⁴ Die Inseratsübersichten lassen sich nach Be- und Entladeort, nach Postleitzahlen und Postleitzahlbereichen, nach inserierendem Unternehmen sowie über eine Volltextsuche filtern.⁴¹⁵ Zur einfacheren Erstellung mehrerer Inserate in einem Arbeitsschritt bietet der Softwareprototyp die Möglichkeit, Inserate per Excel-Tabelle hochzuladen.⁴¹⁶

⁴⁰⁹ Vgl. FÖHRING/BRUNS (2011), S. 49.

⁴¹⁰ Vgl. FÖHRING/BRUNS (2011), S. 49, sowie KLIPPERT et al. (2011), S. 53, 55 u. 77.

⁴¹¹ Vgl. FÖHRING et al. (2012), S. 29 ff. u. 48 ff., sowie FÖHRING/BRUNS (2011), S. 36.

⁴¹² Vgl. FÖHRING/BRUNS (2011), S. 36 f.

⁴¹³ Vgl. FÖHRING et al. (2012), S. 36 ff. u. 56.

⁴¹⁴ Vgl. FÖHRING et al. (2012), S. 35.

⁴¹⁵ Vgl. FÖHRING et al. (2012), S. 42 ff. u. 56.

⁴¹⁶ Vgl. FÖHRING et al. (2012), S. 54 ff.

Angebote [Zu Nachfragen wechseln](#)

ALLE ANGEBOTE

Hier sehen Sie alle Angebote.


















NUMMER	STARTORT (BF)	ZIELORT (BF)	AB DATUM	ZEITRAUM	UNTERNEHMEN
 #A.000.448	Stuttgart	Karlsruhe	22.11.2013	wöchentlich	Burghardt AG
 #A.000.446	Hagen	Wiesbaden	20.09.2013	einmalig	Schmit AG
 #A.000.447	Stuttgart	Paderborn	07.11.2013	jährlich	Fröhlich GmbH
 #A.000.444	Stuttgart	Hagen	18.01.2013	jährlich	Jackle GmbH
 #A.000.445	Karlsruhe	Paderborn	20.03.2013	einmalig	Enders GmbH
 #A.000.441	Stuttgart	Karlsruhe	21.09.2013	jährlich	Kretschmer AG
 #A.000.442	Essen	Augsburg	05.03.2013	wöchentlich	Enders GmbH
 #A.000.439	Paderborn	Stuttgart	14.10.2013	einmalig	Ferstl AG
 #A.000.440	Karlsruhe	Karlsruhe	08.06.2013	monatlich	Glocke GmbH
 #A.000.438	Wiesbaden	Stuttgart	15.07.2013	monatlich	Schmit AG
 #A.000.435	Hagen	Hagen	08.11.2013	jährlich	Schall AG
 #A.000.433	Bremen	Hagen	24.03.2013	wöchentlich	Neher AG
 #A.000.434	Bremen	Essen	21.03.2013	wöchentlich	Grote AG
 #A.000.430	Essen	Paderborn	08.11.2013	jährlich	Neher AG
 #A.000.431	Karlsruhe	Wiesbaden	08.06.2013	wöchentlich	Hinz GmbH
 #A.000.432	Karlsruhe	Essen	02.03.2013	monatlich	Lüke GmbH
 #A.000.428	Stuttgart	Hagen	25.05.2013	monatlich	Petersen AG

Abbildung 3: Übersicht über vorhandene Transportangebote⁴¹⁷⁴¹⁷ Quelle: eigene Darstellung.

Neues Inserat für ein Transportangebot erstellen

NEUES INSERAT FÜR EIN TRANSPORTANGEBOT ERSTELLEN

Die mit einem (*) gekennzeichneten Felder sind Pflichtfelder.

Basisdaten:

Unternehmen mit einem Transportangebot*

Inserat gültig ab*
(Format: TT.MM.JJJJ)

Inserat gültig bis*
(Format: TT.MM.JJJJ)

Zeitraum
Frequenz*

Beladestelle für Fracht:

Gleisanschluss vorhanden? ☒ Ja ☐ Nein

Beladebahnhof
Suche nach Nr oder Name

Unternehmensname

Straße

(fortgesetzt)

PLZ

Stadt

Land

Entladestelle für Fracht:

Gleisanschluss vorhanden? ☒ Ja ☐ Nein

Entladebahnhof
Suche nach Nr oder Name

Unternehmensname

Straße

(fortgesetzt)

PLZ

Stadt

Land

Transportdaten

Angebot umfasst* ☒ reine Traktion (nur Lokomotiven) ☐ Traktion und Güterwagen

Sonstige Informationen:

Kontaktperson

Unternehmenskategorien:

☐ Eisenbahnverkehrsunternehmen

☐ Eisenbahninspektion

☐ verladende Unternehmen

☐

Abbildung 4: Eingabemaske für ein Transportangebot⁴¹⁸⁴¹⁸ Quelle: eigene Darstellung.

Inserat #A.000.374

Inserat verhandeln
Mit Nachfrage beantworten
E-Mail an Anbieter

UNTERNEHMEN MIT EINEM TRANSPORTANGEBOT: FRAUENBERG

Inserat gültig bis 16.02.2013

Versatel GmbH BELADESTELLE FÜR FRACHT Frauenberg Nauheimer Str. 36 55776 Frauenberg Germany	Lehnen AG ENTLADESTELLE FÜR FRACHT Jagsthausen Bulderweg 144 74249 Jagsthausen Germany
Kein Gleisanschluss vorhanden.	Kein Gleisanschluss vorhanden.

Frequenz	jährlich
Verfügbar ab	12.02.2013
Häufigkeit für wiederholte Transporte	1

LADERAUMBESCHREIBUNG

Angebot umfasst:	reine Traktion (nur Lokomotiven)
------------------	----------------------------------

UNTERNEHMENSPROFIL
Anzeigen

Berger GmbH

PLZ	82380
Stadt	Peißenberg
Land	Deutschland
Telefon	011-97-43-4930
E-Mail	vjones@privacy.net
Internet	http://wolf.net
Handelsregister-Eintragung	-
Umsatzsteuer-Identifikationsnummer	-

Inserate anzeigen

Abbildung 5: Detailansicht eines Transportangebots⁴¹⁹

Es wurden technische Schnittstellen geschaffen, um eine Anbindung an Drittanbietersoftware zu ermöglichen.⁴²⁰

Zur Generierung von Nutzungsstatistiken wird das Verhalten der Nutzer innerhalb des Softwareprototyps möglichst vollständig erfasst.⁴²¹ Mit Hilfe der Nutzungsstatistiken können die zuständigen Mitarbeiter eines registrierten Unternehmens und die Betreiber der Online-Frachtenbörse das Verhalten ihrer Nutzer nachvollziehen.⁴²²

Nutzer des Softwareprototyps können andere Nutzer bewerten.⁴²³ Die Bewertungen sind rein natürlichsprachlicher Art und enthalten keine quantitative Merkmale.⁴²⁴ Jedes bewertete Unternehmen entscheidet über die Veröffentlichung seiner Bewertung auf seinem Unternehmensprofil.⁴²⁵

⁴¹⁹ Quelle: eigene Darstellung.

⁴²⁰ Vgl. FÖHRING/BRUNS (2011), S. 39 ff.

⁴²¹ Vgl. FÖHRING/BRUNS (2011), S. 46.

⁴²² Vgl. FÖHRING/BRUNS (2011), S. 46.

⁴²³ Vgl. FÖHRING et al. (2012), S. 58 ff., sowie FÖHRING/BRUNS (2011), S. 50.

⁴²⁴ Vgl. FÖHRING et al. (2012), S. 58 ff., sowie FÖHRING/BRUNS (2011), S. 50.

⁴²⁵ Vgl. FÖHRING et al. (2012), S. 58 ff., sowie FÖHRING/BRUNS (2011), S. 50.

Neben den Bewertungsmöglichkeiten für Nutzer untereinander, die zur Feedbackanalyse herangezogen werden können, bietet der Softwareprototyp ein Kontaktformular.⁴²⁶ Über dieses Formular können sich Nutzer mit einem Freitext zur Online-Frachtenbörse äußern und direkt mit den Betreibern in Kontakt treten.⁴²⁷

Nutzer des Softwareprototyps können die Inserate anderer Unternehmen hervorheben.⁴²⁸ Analog lassen sich die Inserate anderer Unternehmen gezielt ignorieren.⁴²⁹

Zusätzlich kann jeder Nutzer individuelle Regeln definieren, um benachrichtigt zu werden, wenn ein Inserat eingestellt wird, welches für ihn interessant sein könnte.⁴³⁰ Diese Regeln setzen sich aus beliebig vielen Konditionen zusammen. Sind alle Konditionen einer Regel erfüllt, gilt das Inserat als interessant.⁴³¹ Bei Erstellung eines neuen Inserats prüft der Softwareprototyp für alle Nutzer, ob das betreffende Inserat eine ihrer Regeln erfüllt.⁴³² Ist dies der Fall, erhalten die betreffenden Nutzer eine Benachrichtigung über das Inserat im System des Softwareprototyps.⁴³³ Diese Benachrichtigungen können per E-Mail abonniert werden.⁴³⁴

Die Datenbank des Softwareprototyps kann mit einer Liste europäischer Bahnhöfe initialisiert werden.⁴³⁵ Diese sowie alle später eingepflegten Bahnhöfe können in einem Bahnhofsverzeichnis von den Nutzern eingesehen werden.⁴³⁶

Zudem lassen sich alle im System des Softwareprototyps registrierten Unternehmen in einem Branchenbuch einsehen.⁴³⁷

2.4.7 Kritische Reflexion der bisherigen Entwicklung

Der in den Jahren 2010 bis 2012 implementierte Softwareprototyp einer Online-Frachtenbörse konnte nicht alle Anforderungen berücksichtigen, die von verschiedensten Stakeholdern des CODE24-Projekts an ihn gestellt wurden. Der primäre Grund hierfür ist

⁴²⁶ Vgl. FÖHRING et al. (2012), S. 76, sowie FÖHRING/BRUNS (2011), S. 52.

⁴²⁷ Der Softwareprototyp generiert bei Nutzung des Kontaktformulars eine E-Mail aus dem vom Nutzer eingegebenen Text, welche zusammen mit seinen Kontaktinformationen direkt an die Kundendienst-Mitarbeiter der Online-Frachtenbörse geschickt wird. Vgl. FÖHRING/BRUNS (2011), S. 52.

⁴²⁸ Vgl. FÖHRING et al. (2012), S. 65 f.

⁴²⁹ Vgl. FÖHRING et al. (2012), S. 66 f.

⁴³⁰ Vgl. FÖHRING et al. (2012), S. 70 ff.

⁴³¹ Vgl. FÖHRING et al. (2012), S. 72 ff.

⁴³² Vgl. FÖHRING et al. (2012), S. 70.

⁴³³ Vgl. FÖHRING et al. (2012), S. 70.

⁴³⁴ Vgl. FÖHRING et al. (2012), S. 70.

⁴³⁵ Vgl. FÖHRING et al. (2012), S. 19 f.

⁴³⁶ Vgl. FÖHRING et al. (2012), S. 75.

⁴³⁷ Vgl. FÖHRING et al. (2012), S. 67 ff.

in der Tatsache zu sehen, dass die Implementierung den zeitlichen und finanziellen Restriktionen einer Diplomarbeit unterlag und viele der von hierzu befragten Branchenexperten geäußerten Anforderungen über die von der Wissenschaft leistbare Arbeit hinausgingen.⁴³⁸

Als Erfolg ist die Tatsache zu werten, dass im Jahr 2013 zwei Online-Frachtenbörsen für den Schienengüterverkehr gestartet sind, welche auf dem in diesem Kapitel vorgestellten Softwareprototyp basieren.⁴³⁹

Im Folgenden wird dennoch ein kritischer Überblick darüber gegeben, welche der bisher identifizierten Anforderungen durch den implementierten Softwareprototyp nicht abgedeckt wurden. Es werden zunächst jene Defizite behandelt, die aus den erwähnten Restriktionen der Implementierung resultieren. Im Anschluss werden Anforderungen diskutiert, welche bewusst nicht implementiert wurden.

Defizite des vorliegenden Softwareprototyps

Die Eingabemasken zur Erstellung von Inseraten zu Transportangeboten und -nachfragen wurden anhand der Angaben von Branchenexperten entwickelt.⁴⁴⁰ Es fehlt jedoch an einer informationstechnischen Abbildung der realen Gegebenheiten auf Strecken und in Haltestellen, wie bspw. Terminals und Bahnhöfen.⁴⁴¹ Derartige kriterielle Besonderheiten quantitativ zu erfassen könnte dazu beitragen, zu einer qualitativ besseren Vermittlung von Angebot und Nachfrage innerhalb der Online-Frachtenbörse zu kommen.⁴⁴²

Die Bewertung von Geschäftspartnern wurde lediglich grundlegend implementiert.⁴⁴³ Die Nutzer haben die Möglichkeit, sich mit einem Freitext zu anderen Nutzern zu äußern und diese auf diesem Wege natürlichsprachlich zu bewerten.⁴⁴⁴ Eine quantitative Bewertung findet nicht statt.⁴⁴⁵

Die implementierten Schnittstellen zum Matching von Angebot und Nachfrage, zur Suche nach Inseraten sowie zur Anbindung von Drittanbietersoftware sind funktional.⁴⁴⁶ Jedoch sind sie nur als strukturelle Konzepte angelegt.⁴⁴⁷ So ist bspw. das Einlesen bestehender Daten aus vorhandenen ERP- und WW-Systemen nicht möglich.⁴⁴⁸ Sie sind damit

⁴³⁸ Vgl. FÖHRING/BRUNS (2011), S. 69.

⁴³⁹ Vgl. DÖRR/ENDEMANN (2014), S. 61 f.

⁴⁴⁰ Vgl. FÖHRING/BRUNS (2011), S. 70.

⁴⁴¹ Vgl. FÖHRING/BRUNS (2011), S. 70.

⁴⁴² Vgl. FÖHRING/BRUNS (2011), S. 70.

⁴⁴³ Vgl. FÖHRING/BRUNS (2011), S. 50.

⁴⁴⁴ Vgl. FÖHRING/BRUNS (2011), S. 50.

⁴⁴⁵ Vgl. FÖHRING/BRUNS (2011), S. 70.

⁴⁴⁶ Vgl. FÖHRING/BRUNS (2011), S. 70 f.

⁴⁴⁷ Vgl. FÖHRING/BRUNS (2011), S. 70 f.

⁴⁴⁸ Vgl. FÖHRING/BRUNS (2011), S. 71.

nicht als marktreif anzusehen und entsprechen nicht den Anforderungen, welche bei einem professionellen Produkt zum Einsatz kommen sollten.⁴⁴⁹

Das implementierte Rechtesystem ermöglicht das Zuweisen von Benutzerrollen an Nutzer, welche hierdurch individuelle Berechtigungen der jeweiligen Rolle erhalten. Unternehmen können ebenfalls mit Rollen ausgestattet werden, um ihre Zugehörigkeit zu verschiedenen Gruppen zu definieren.⁴⁵⁰ Mit Hilfe dieses Rechtesystem wäre es möglich, Benutzerrollen zu erstellen, welche den Betreibern der Online-Frachtenbörse ermöglichen würden, ihre Nutzer in A-, B- und C-Gruppen einzuteilen, einzelnen Nutzern verschiedene Bearbeitungszustände zuzuweisen oder geschlossene Gruppen zu realisieren. In der vorliegenden Implementierung wurde lediglich der letzte Punkt auf Unternehmensebene realisiert, um einzelne Inserate nur für Unternehmen sichtbar zu machen, welche bestimmten Gruppen angehören.⁴⁵¹

Kritische Anforderungen

Insgesamt lässt sich in der Fachliteratur eine Tendenz erkennen, dass die befragten Experten die von ihnen avisierte Online-Frachtenbörse für den Schienengüterverkehr nicht nur als Handelsplatz, sondern auch als Informationssystem – als eine „Datendrehscheibe im Internet“ – sehen.⁴⁵²

Diese Äußerung muss – ebenso wie alle im Folgenden kritisierten Anforderungen – vor dem Hintergrund reflektiert werden, dass eine Erhebung von BRUNS et al. ergab, dass zwei Drittel der Experten den potentiellen Nutzern entweder keine oder eine nur sehr geringe Zahlungsbereitschaft für die Nutzung von Online-Frachtenbörsen von maximal 100 EUR pro Jahr attestierten.⁴⁵³

Abbildung 6 zeigt die Ergebnisse dieser Erhebung.

⁴⁴⁹ Vgl. FÖHRING/BRUNS (2011), S. 70 f.

⁴⁵⁰ In der vorliegenden Implementierung nehmen die Unternehmen diese Zuweisung selbst vor. Vgl. FÖHRING et al. (2012).

⁴⁵¹ Vgl. FÖHRING et al. (2012).

⁴⁵² Vgl. ENDEMANN/KASPERS (2011), S. 35.

⁴⁵³ Auf die Frage nach der Zahlungsbereitschaft für die Nutzung von Online-Frachtenbörsen haben 6 Experten geantwortet. Die Hälfte gab an, keine Zahlungsbereitschaft zur Nutzung einer Online-Frachtenbörse zu sehen. Die weiteren drei Experten sprachen sich jeweils für eine geringe Grundgebühr (bis 100 EUR p.a.), eine Fixgebühr pro Transaktion bzw. einen geringen festen Nutzungsbeitrag in Kombination mit einer transaktionsabhängigen Vermittlungsgebühr aus. Vgl. BRUNS et al. (2010b), S. 13.

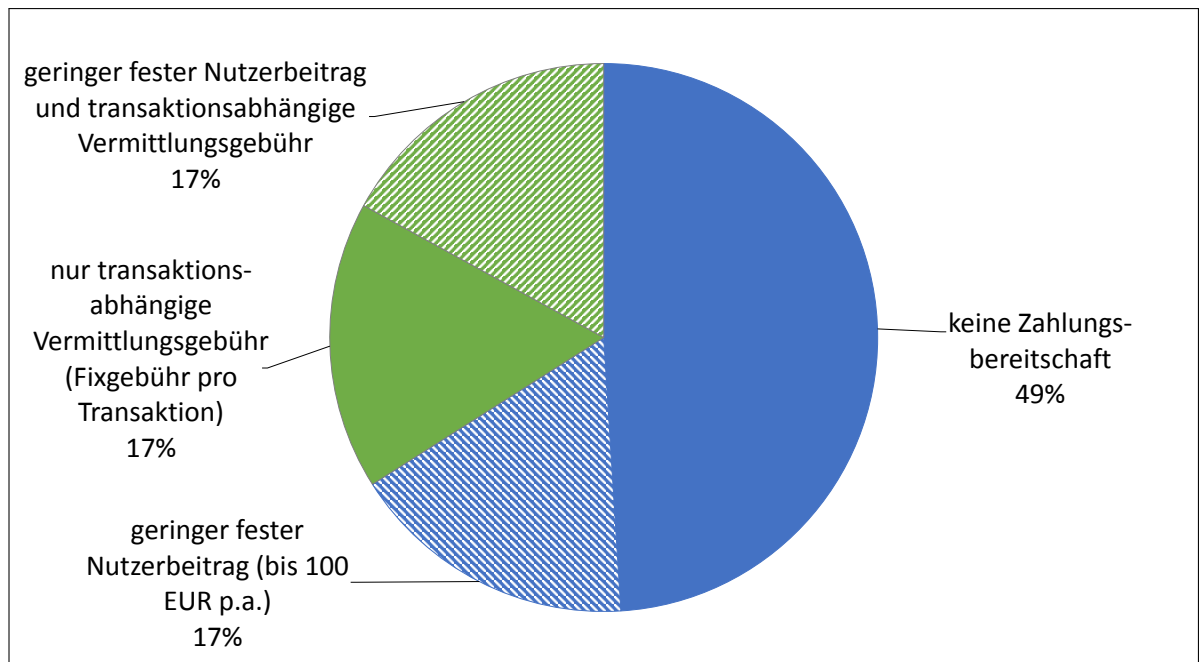


Abbildung 6: Zahlungsbereitschaft der potentiellen Nutzer einer Online-Frachtenbörse⁴⁵⁴

Vor dem Hintergrund des Projekts CODE24 wird im Rahmen der Unterstützung multimodaler Verkehre gefordert, andere Verkehrsträger neben dem Verkehrsträger Schiene zu unterstützen, allerdings nur im Vor- oder Nachlauf zur Schiene. Während diese Forderung im Rahmen des Projektziels der Verlagerung von Güterverkehren von der Straße auf die Schiene legitim erscheint, verliert sich ihre politische Bedeutung bei der Konzeption einer unabhängig vom Projekt CODE24 gedachten Online-Frachtenbörse für den europäischen Güterverkehr.

Es wird einerseits gefordert, dass potentielle Geschäftspartner als Vorauswahl aufgelistet und Nutzer zeitnah darüber informiert werden sollen, wenn ein neu registrierter Nutzer im System der Online-Frachtenbörse als potentieller Geschäftspartner identifiziert wird.⁴⁵⁵ Andererseits wird gefordert, dass Anbieter für ihre Wettbewerber nicht sichtbar sein dürfen.⁴⁵⁶ Dies führt zu einem Konflikt, da potentielle Geschäftspartner im Rahmen einer Online-Frachtenbörse auch gleichzeitig Wettbewerber sein können, da in der Fachliteratur ebenfalls gefordert wird, dass Eisenbahnverkehrsunternehmen über die Online-Frachtenbörse andere Eisenbahnverkehrsunternehmen mit der Durchführung eines Gütertransports beauftragen können sollen.⁴⁵⁷

Einige in der Fachliteratur geäußerten Anforderungen an eine Erfolg versprechende Online-Frachtenbörse sind als problematisch zu betrachten, da der mit ihrer Realisierung verbundene Aufwand als hoch einzustufen ist. Neben ihrer Kernleistung, der Vermittlung

⁴⁵⁴ Quelle: eigene Darstellung in Anlehnung an BRUNS et al. (2010b), S. 13.

⁴⁵⁵ Vgl. KLIPPERT et al. (2011), S. 76 u. 79 f.

⁴⁵⁶ Vgl. HABIB/BRUNS (2012), S. 38.

⁴⁵⁷ Vgl. HABIB/BRUNS (2012), S. 42.

von Transportdienstleistungen, solle die Online-Frachtenbörse zusätzliche Leistungen wie die Anmietung von Wagen, die Vermittlung von Personaldienstleistungen sowie direkte Reservierung von KV-Terminals anbieten.⁴⁵⁸ Dabei solle im Fall der Reservierung von KV-Terminals aus der Online-Frachtenbörse ersichtlich sein, welche Terminals zu welchen Zeiten bereits belegt sind.⁴⁵⁹ Zusätzlich zu diesen Services solle die Online-Frachtenbörse einen Reklamations- und Inkassoservice anbieten.⁴⁶⁰ Auch sollten alle Besonderheiten, welche einen Transport beeinflussen könnten, wie bspw. Baustellen und Verkehrsunfälle, zeitnah in der Online-Frachtenbörse ersichtlich sein.⁴⁶¹ Für den Fall, dass ein Nutzer Fragen zu einem Transport haben sollte, wird die Existenz eines telefonischen Kundendienst-Hotline gefordert.⁴⁶² Die Forderung der Experten nach einem telefonischen Kundendienst für die Nutzer der Online-Frachtenbörse ist vor dem Hintergrund der sehr geringen Zahlungsbereitschaft bereits ohne weitere Forderungen problematisch, da telefonischer Kundendienst bspw. im Gegensatz zu Kundendienst per E-Mail mit höherem Aufwand verbunden ist.⁴⁶³ Die Fachliteratur regt jedoch an, dass der telefonische Kundendienst möglichst rund um die Uhr erreichbar sein und die Kundendienst-Hotline möglichst kostenfrei zur Verfügung gestellt werden soll.⁴⁶⁴

Bereits die Realisierung einer Auswahl der hier kritisch hinterfragten Anforderungen würde den Kapitalbedarf einer Online-Frachtenbörse signifikant erhöhen, da der mit ihrer Realisierung verbundene Aufwand als hoch einzustufen ist. Vor dem Hintergrund einer allgemein als niedrig eingestuften Zahlungsbereitschaft erscheint dies nicht möglich.

⁴⁵⁸ Vgl. HABIB/BRUNS (2012), S. 42, ENDEMANN/KASPERS (2011), S. 35, sowie KLIPPERT et al. (2011), S. 67 u. 76.

⁴⁵⁹ Vgl. HABIB/BRUNS (2012), S. 42.

⁴⁶⁰ Vgl. HABIB/BRUNS (2012), S. 14.

⁴⁶¹ Vgl. HABIB/BRUNS (2012), S. 44.

⁴⁶² Vgl. HABIB/BRUNS (2012), S. 45.

⁴⁶³ Dies beruht auf der Plausibilitätsüberlegung, dass das Medium E-Mail eine asynchrone Kommunikationsführung ermöglicht und dem Kundendienst eine Priorisierung und arbeitsteilige Erledigung seiner Aufgaben ermöglicht. Dies kann aufgrund der Asynchronität zu vordefinierten Zeitpunkten erfolgen. So könnten die Betreiber einer Online-Frachtenbörse den Kundendienst in einem vordefinierten Zeitfenster, bspw. täglich von 10 bis 14 Uhr, mit Hilfe von Teilzeitkräften durchführen. Die Kommunikation über Medien wie Telefon, Text- oder Video-Chat erfordert im Gegensatz hierzu eine unmittelbare Reaktion seitens des Kundendienstes und setzt im Extremfall die weiter unten geforderte, unterbrechungsfreie Erreichbarkeit desselben voraus.

⁴⁶⁴ Vgl. HABIB/BRUNS (2012), S. 14 u. 45.

3 Betriebswirtschaftliche Desiderata im Schienengüterverkehr

Aus den geschilderten Problemen und Anforderungen verschiedener Stakeholder lassen sich vier, für die Bestimmung der betriebswirtschaftlichen Desiderata relevante, Realprobleme ableiten, welche der Etablierung einer zentralen Online-Frachtenbörse für den Schienengüterverkehr entgegenstehen.

Das erste Realproblem ist der verlustfreie Betrieb einer Online-Frachtenbörse für schienengebundene Güterverkehre. Die durch den Betrieb der Online-Frachtenbörse entstehenden Kosten müssen über zu erwirtschaftende Umsätze gedeckt werden. Hierfür muss ein geeignetes Geschäftsmodell gefunden werden, welches für jeden Teilnehmer der Online-Frachtenbörse eine Nutzungsgebühr festlegt.⁴⁶⁵ Die Nutzung der Online-Frachtenbörse müsste wiederum aus Teilnehmersicht wirtschaftliche Vorteile bieten, um die Teilnahmekosten in Form dieser Nutzungsgebühr zu rechtfertigen.

Das zweite Realproblem stellt die Weitergabe wettbewerbssensitiver Daten dar. Alle Teilnehmer eines zentral organisierten Marktplatzes sind gezwungen, ihre Daten an den zentralen Betreiber zu übermitteln. Dies setzt ein hohes Vertrauen in die Diskretion des Betreibers voraus. Hieraus ergibt sich die Forderung der Wettbewerbsneutralität des Betreibers.

Das dritte Realproblem stellt die Erfahrung des potentiellen Betreibers dar. Die Rolle des Betreibers einer Online-Frachtenbörse zur Versteigerung von Transportaufträgen erfordert eine intime Kenntnis der jeweiligen Branche. Gleichzeitig fordern hierzu befragte potentielle Marktplatzteilnehmer⁴⁶⁶ die Neutralität des Betreibers im Markt der schienengebundenen Güterverkehre.⁴⁶⁷ Aufgrund der geringen Marktgröße und -fragmentierung⁴⁶⁸ gestaltet es sich schwierig, einen Betreiber zu finden, der die nötige Erfahrung im Bereich schienengebundener Güterverkehre besitzt, aber nicht in der einen oder anderen Form selbst Teilnehmer dieses Marktes ist.

Das vierte Realproblem stellt die Berücksichtigung des kombinierten Verkehrs dar. Da der kombinierte Verkehr in Europa zukünftig eine größere Rolle spielen wird⁴⁶⁹, findet sich in Veröffentlichungen zu den Erfordernissen einer Online-Frachtenbörse regelmäßig

⁴⁶⁵ Für den Markt straßengebundener Güterverkehre haben BRUNS et al. zwei Geschäftsmodelle identifiziert: Das „transportvolumen- und transaktionsabhängige“ sowie das „nutzer- und zeitabhängige“ Geschäftsmodell. Vgl. BRUNS et al. (2012a), S. 7.

⁴⁶⁶ Vgl. HABIB/BRUNS (2012), S. 22 f.

⁴⁶⁷ Vgl. HABIB/BRUNS (2012), S. 38 f.

⁴⁶⁸ BRUNS et al. beziffern die potentielle Nutzeranzahl einer Online-Frachtenbörse in Deutschland auf 5 bis 60 Eisenbahnverkehrsunternehmen, 225 bis 2.700 Expeditionen sowie ca. 7.000 bis 83.000 Hersteller. Vgl. BRUNS et al. (2012a), S. 15.

⁴⁶⁹ Vgl. ICKERT et al. (2007), S. 61 f. u. 99.

die Forderung einer Berücksichtigung von kombinierten Verkehren – in der Hauptsache mit einem Vor- und Nachlauf per Lkw⁴⁷⁰, einem Hauptlauf mittels Eisenbahn oder Schiff sowie der Einbeziehung von Infrastrukturbetreibern von KV-Terminals⁴⁷¹ – wieder.⁴⁷²

Die Herausforderung bei der Lösung des ersten und zweiten Realproblems besteht in der Minimierung der Betriebs- und Teilnahmekosten der potentiellen Online-Frachtenbörse bzw. der möglichst glaubhaften Garantie der Neutralität und Diskretion des Betreibers dieses elektronischen Marktplatzes. Aus den beschriebenen vier Realproblemen lassen sich einige betriebswirtschaftliche Desiderata ableiten.

Wünschenswert ist die Existenz eines Marktplatzes, welcher ...

1. der Versteigerung von Transportdienstleistungen für den Hauptlauf im Schienengüterverkehr dient,
2. von einer neutralen Partei, die mit den Erfordernissen des Verkehrsträgers Schiene vertraut ist, verlustfrei betrieben werden kann,
3. die Einstiegshürden für seine potentiellen Nutzer⁴⁷³ so niedrig wie möglich hält,⁴⁷⁴
4. die Überwindung von Informationsasymmetrien und -defiziten fördert sowie
5. den Vor- und Nachlauf über andere Verkehrsträger im kombinierten Verkehr unterstützt.

Es wird deutlich, dass eine Erfolg versprechende Online-Frachtenbörse für den Schienengüterverkehr eine multimodale Frachtenbörse sein muss, welche in der Lage ist, verschiedenartige Verkehrsträger zu berücksichtigen, um die Durchführung kombinierter Verkehre zu ermöglichen. Dies legt den Schluss nahe, dass ein Konzept für eine über rein schienengebundene Transporte hinausgehende Online-Frachtenbörse betriebswirtschaftlich wünschenswert ist.

Weiterhin wird deutlich, dass die ersten drei Realprobleme auf die *zentrale* Natur der potentiellen Online-Frachtenbörse zurückzuführen sind, da bei einer zentralen Realisierung der Online-Frachtenbörse ein einzelner Betreiber die Kosten zur Bereitstellung der Infrastruktur des Marktplatzes tragen und über die Daten aller Teilnehmer verfügen würde sowie zum Zwecke der Neukundenakquisition potentiellen Teilnehmern seine Expertise bzgl. der Domäne des von ihm betriebenen Marktplatzes versichern müsste. Zudem erscheint

⁴⁷⁰ Vgl. HABIB/BRUNS (2012), S. 26 u. 29, sowie ENDEMANN/KASPERS (2011), S. 32.

⁴⁷¹ Vgl. HABIB/BRUNS (2012), S. 42, sowie ENDEMANN/KASPERS (2011), S. 32 u. 35.

⁴⁷² Vgl. HABIB/BRUNS (2012), S. 41, sowie KLIPPERT et al. (2011), S. 75.

⁴⁷³ Im Schienengüterverkehr sind dies Verlader, Spediteure und Eisenbahnverkehrsunternehmen. Vgl. ENDEMANN/KASPERS (2011), S. 33.

⁴⁷⁴ So sollten aus Teilnehmersicht bspw. möglichst keine oder nur geringe Kosten zur Nutzung anfallen. Diese Anforderung steht in Konflikt mit dem Gewinnmaximierungsziel des potentiellen Betreibers der Online-Frachtenbörse.

es plausibel, dass die Teilnehmer sich sowohl bei der erstmaligen Registrierung als auch regelmäßig während der Eingabe von Angeboten von und Nachfragen nach Transportdienstleistungen entscheiden müssten, ob sie diese sensiblen Daten zugänglich machen möchten.

Es wäre zu erforschen, ob ein *dezentral* automatisierter Ansatz bei der Realisierung einer Online-Frachtenbörse als betriebswirtschaftlich attraktive Alternative zu dem bisher verfolgten zentralen Ansatz anzusehen wäre.

Diese Alternative würde die zuvor genannten vier Realprobleme abschwächen, da ...

1. kein einzelner, zentraler Betreiber benötigt würde,⁴⁷⁵
2. alle Teilnehmer die gleichen Daten zur Verfügung hätten,⁴⁷⁶
3. aus Teilnehmersicht nur sehr geringe Kosten anfallen würden⁴⁷⁷ sowie
4. die Berücksichtigung kombinierter Verkehre in einer automatisierten Online-Frachtenbörse erheblich leichter zu realisieren wäre.⁴⁷⁸

Dies legt den Schluss nahe, dass ein Konzept für eine dezentrale, automatisierte Online-Frachtenbörse betriebswirtschaftlich wünschenswert ist.

⁴⁷⁵ Bei einer dezentralen Realisierung der Online-Frachtenbörse könnten alle Teilnehmer gemeinsam die Infrastruktur bereitstellen. Jeder Teilnehmer würde mit einem Computer einen Knoten eines Agentennetzwerks bilden. Die Betreiberkosten der Online-Frachtenbörse würden so gegenüber einer zentral betriebenen Variante sinken, da alle Teilnehmer diese Kosten gemeinsam tragen würden.

⁴⁷⁶ Diese wettbewerbssensiblen Daten wären in einer softwaretechnischen Implementierung der Online-Frachtenbörse verschlüsselt. Dennoch lässt sich feststellen, dass alle Teilnehmer über den gleichen Datensatz verfügen würden, da jeder Teilnehmer seine Daten automatisch mit jedem anderen Teilnehmer austauscht. Somit müssten im Gegensatz zu einer zentral betriebenen Online-Frachtenbörse keine Maßnahmen zur Regelung des Informationszugriffs ergriffen werden.

⁴⁷⁷ Jedes am dezentralen Agentennetzwerk teilnehmende Unternehmen müsste die Software zur Teilnahme an der Online-Frachtenbörse auf einem internetfähigen Computer installieren und diesen zur Teilnahme eingeschaltet lassen, wodurch Kosten entstehen würden.

⁴⁷⁸ Die Frage, ob die Online-Frachtenbörse zentral oder dezentral organisiert ist, bleibt hiervon unberührt. Der Grund für die leichtere Berücksichtigung von kombinierten Verkehren in einer automatisierten Online-Frachtenbörse gegenüber einer nicht-automatisierten Online-Frachtenbörse ist, dass automatisierte Prozesse leichter kombinatorische Probleme lösen können als dies mit manuellen Eingabemasken zu realisieren wäre, da die Komplexität einer Eingabemaske zur Berücksichtigung kombinierter Verkehre deutlich steigen würde.

Es erscheint darüber hinaus naheliegend, dass ein Konzept wünschenswert wäre, welches eine effiziente⁴⁷⁹ Versteigerung der gehandelten Transportdienstleistungen gewährleistet. Entscheidend für die Effizienz der Versteigerung ist die Wahl der Auktionsform.⁴⁸⁰

Der Einsatz der zweiseitigen Auktionsform erscheint aus zwei Gründen adäquat: Zum einen sind viele große Aktienbörsen und Ressourcenmärkte in der realen Welt als zweiseitige Auktionen organisiert.⁴⁸¹ Zum anderen erhalten die Teilnehmer auf diese Weise keine dedizierten Rollen zugewiesen, sondern agieren ihren jeweiligen Präferenzen entsprechend in der Rolle als Anbieter oder Nachfrager von Transportdienstleistungen.

Neben dem Konzept für eine dezentrale, automatisierte Online-Frachtenbörse erscheint eine prototypische Implementierung dieses Konzepts in Form einer Software betriebswirtschaftlich wünschenswert. Ein solcher Softwareprototyp könnte die weitere Forschung zum Thema „Online-Frachtenbörse für den Schienengüterverkehr“ unterstützen und in die Lage versetzen, die skizzierten Konzepte sowie die ihnen zugrundeliegenden Ideen und Funktionen für eine interessierte Zielgruppe im Rahmen weiterer Anforderungsanalysen und Marktstudien tatsächlich erlebbar zu machen. Der Softwareprototyp sollte demnach in der Lage sein, die weitere Forschung in Richtung einer dezentralen, automatisierten Online-Frachtenbörse zu unterstützen. Zu diesem Zweck muss die Nutzung des Softwareprototyps auch ohne IT-Kenntnisse möglich sein, da die Interessengruppen aus einem nicht IT-affinen Umfeld kommen. Der Softwareprototyp sollte zudem die technische Machbarkeit der entwickelten Konzepte demonstrieren.

⁴⁷⁹ Bei der Durchführung von Auktionen wird zwischen optimalen und effizienten Auktionsformen unterschieden. Optimale Auktionsformen maximieren den Gewinn des Auktionators, während effiziente Auktionsformen zu einer wohlfahrtsmaximierenden Allokation führen, bei der die versteigerten Güter jenen Bietern mit der höchsten Bewertung für das jeweilige Gut zufallen und nicht jenen, welche das höchste Gebot abgegeben haben. KRISHNA und PERRY beschreiben diese Differenzierung vor dem Hintergrund von Multi-Objekt-Auktionen. Vgl. KRISHNA/PERRY (1998), S. 3 f.

⁴⁸⁰ AUSUBEL und CRAMTON beschreiben, dass eine Auktion, die sich im Sinne der Erläuterung aus Fußnote 479 als optimal erweist, nicht zwangsläufig zur effizienten Verteilung der versteigerten Güter führt. Vgl. AUSUBEL/CRAMTON (1998), S. 2 ff.

⁴⁸¹ YANG verwendet dieses Argument zur Rechtfertigung des Einsatzes zweiseitiger Auktionen in computergestützten Simulationen. Vgl. YANG (2002), S. 92.

4 Wissenschaftliche Probleme

In der Fachliteratur wird dem Thema „Online-Frachtenbörse für Transportleistungen im Schienengüterverkehr“ bisher wenig Beachtung geschenkt.⁴⁸²

Der Einsatz von zweiseitigen kombinatorischen Auktionen wird dagegen in der Fachliteratur ebenso ausführlich für verschiedene Märkte diskutiert⁴⁸³ wie der Einsatz von Multi-Agenten-Systemen⁴⁸⁴.

Die Zusammenführung der Erkenntnisse über die Erfordernisse einer Online-Frachtenbörse für Transportdienstleistungen im Schienengüterverkehr, über den Einsatz zweiseitiger kombinatorischer Auktionen sowie über die Gestaltung autonomer Multi-Agenten-Systeme konstituiert das erste wissenschaftliche Problem dieser Forschungsarbeit.

Das zweite wissenschaftliche Problem besteht in der Konzeption und Implementierung eines prototypischen Systems autonomer, dezentral organisierter Software-Agenten für die Durchführung zweiseitiger kombinatorischer Auktionen von multimodalen Transportdienstleistungen im europäischen Güterverkehr am Beispiel des kombinierten Verkehrs, welches in der Lage ist die weitere Forschung im Bereich „Online-Frachtenbörsen für den Schienengüterverkehr“ zu unterstützen.

Aus Gründen der Lesbarkeit wird mit dem Begriff „Softwareprototyp“ von nun an der in dieser Forschungsarbeit konzipierte und implementierte Softwareprototyp bezeichnet. Der in den Jahren 2010 bis 2012 entwickelte Softwareprototyp einer Online-Frachtenbörse für den Schienengüterverkehr wird als „ORFE-Softwareprototyp“ („Online Rail Freight Exchange“) bezeichnet. In Fällen, in denen eine Unterscheidung der beiden Prototypen erforderlich ist, wird der in dieser Forschungsarbeit konzipierte und implementierte Prototyp eines Systems autonomer, dezentral organisierter Software-Agenten als „AFEX-System“ („Agent-Based Freight Exchange“) bezeichnet.

⁴⁸² Der Großteil der Veröffentlichungen zu diesem Thema wurde im Rahmen des Projekts CODE24 am Institut für Produktion und Industrielles Informationsmanagement der Universität Duisburg-Essen publiziert. Vgl. FÖHRING/ZELEWSKI (2013), BRUNS et al. (2012a), BRUNS et al. (2012b), FÖHRING et al. (2012), HABIB/BRUNS (2012), FÖHRING/BRUNS (2011), KLIPPERT et al. (2011), WEICHEL/-BRUNS (2011), BRUNS et al. (2010a), sowie BRUNS et al. (2010b). Darüber hinaus existieren nur wenige Veröffentlichungen, von denen viele eine Online-Frachtenbörse für den Schienengüterverkehr lediglich avisieren. Vgl. ENDEMANN/KASPERS (2011), sowie SCHECK/WILSKE (2011).

⁴⁸³ Vgl. PERENNES (2013), ACKERMANN et al. (2011), YANG (2002), PARKES/UNGAR (2001), sowie RASSENTI et al. (1982).

⁴⁸⁴ Vgl. DAVIDSSON et al. (2005), FOX et al. (2000), JENNINGS (2000), BURMEISTER et al. (1997), sowie WOOLDRIGE/JENNINGS (1995).

5 Begrifflichkeiten und konzeptionelle Grundlagen

5.1 Elektronische Marktplätze

5.1.1 Marktplatzbegriff

Ein Marktplatz bezeichnet eine Institution, die ihren Akteuren erlaubt, sich zu einer bestimmten Zeit an einem bestimmten Ort zu treffen und dort Kauf- sowie Verkaufsabsichten in der Annahme zu äußern, dass die eventuelle Übereinstimmung dieser Absichten zu einer Transaktion von Sachgütern oder Dienstleistungen zwischen Käufer und Verkäufer führt.⁴⁸⁵

Die Akteure treten auf einem Marktplatz als Anbieter oder Nachfrager auf.⁴⁸⁶

Durch Aufkommen moderner Informations- und Kommunikationstechnologien wurden die Restriktionen bezüglich der zeitlichen und örtlichen Gebundenheit des Zusammenkommens von Käufer und Verkäufer jedoch aufgehoben.⁴⁸⁷ Es entstanden *elektronische Marktplätze*, welche jederzeit erreichbar sind.⁴⁸⁸ Sie ermöglichen es Käufern und Verkäufern erstmals unter einer *virtuellen* Adresse zusammenzukommen.⁴⁸⁹

Durch die informationstechnische Analyse der Entwicklungen von Angebot und Nachfrage waren die Betreiber dieser Marktplätze in der Lage, eine automatisierte Vermittlungsleistung zwischen Käufern und Verkäufern anzubieten.⁴⁹⁰ Dieses Matching von Angebot und Nachfrage stellt eine Unterstützung bei jeder einzelnen Transaktion und – als konkrete Vermittlungsleistung – ein unternehmerisches Produkt des Marktplatzbetreibers dar.⁴⁹¹

KOLLMANN identifiziert vier Probleme des realen Handels, welche durch Zuhilfenahme von Informationstechnologien gelöst werden sollten und hierdurch zu einer Zunahme des Einsatzes von elektronischen Marktplätzen führten:⁴⁹²

1. **Kapazitätsbegrenzungen:** In der realen Welt unterliegt ein Marktplatzbetreiber einer räumlichen Restriktion, bspw. innerhalb einer Messehalle.⁴⁹³

⁴⁸⁵ Vgl. GRIEGER (2003), S. 282, sowie KOLLMANN (2001), S. 1 f.

⁴⁸⁶ Vgl. GRIEGER (2003), S. 282, sowie KOLLMANN (2001), S. 2. Die Begriffe „Käufer“ und „Nachfrager“ sowie „Verkäufer“ und „Anbieter“ werden im Folgenden aufgrund dieser Tatsache synonym verwendet.

⁴⁸⁷ Vgl. GRIEGER (2003), S. 282.

⁴⁸⁸ Vgl. GRIEGER (2003), S. 282.

⁴⁸⁹ Vgl. KOLLMANN (2016), S. 457.

⁴⁹⁰ Vgl. KOLLMANN (2016), S. 458.

⁴⁹¹ Vgl. KOLLMANN (2016), S. 458.

⁴⁹² Vgl. KOLLMANN (2016), S. 458.

⁴⁹³ Vgl. KOLLMANN (2016), S. 458.

2. **Vermittlungsrestriktionen:** Betreiber realer Marktplätze beschränken sich in der Regel auf die Bereitstellung von Räumlichkeiten und vermitteln nicht aktiv zwischen Käufern und Verkäufern.⁴⁹⁴
3. **Transparenzdefizite:** In realen Marktplätzen ist es für einen einzelnen Marktteilnehmer lediglich zu hohen Opportunitätskosten möglich, sich eine Marktübersicht zu verschaffen.⁴⁹⁵
4. **Koordinationsineffizienzen:** Käufer und Verkäufer haben in einem realen Marktplatz in der Regel nicht die Möglichkeit, alle Verkäufer bzw. Käufer zu kontaktieren, eine direkte Beziehung zu ihnen zu unterhalten und bei ihren Entscheidungen die Informationen aller Verkäufer bzw. Käufer zu berücksichtigen.⁴⁹⁶

Die folgenden Kapitel zeigen, wie diese Probleme von elektronischen Marktplätzen teilweise gelöst werden.⁴⁹⁷

5.1.2 Der Begriff des elektronischen Marktplatzes

Der Begriff des elektronischen Marktplatzes ist in der Fachliteratur nicht eindeutig definiert.⁴⁹⁸

KOLLMANN beschreibt einen elektronischen Marktplatz als einen nicht-realen Ort, an dem mehrere Anbieter und mehrere Nachfrager – durch einen Marktplatzbetreiber koordiniert – zum Zwecke der Durchführung wirtschaftlicher Transaktionen virtuell zusammenkommen.⁴⁹⁹

Je nach Kontext und Funktionsweise des zugrundeliegenden Marktes kann mit einem elektronischen Marktplatz zudem ein katalogbasiertes System, ein Beschaffungsinstrument, ein interorganisationales Informationssystem oder eine Art Intermediär gemeint sein.⁵⁰⁰ Neben dem Begriff des elektronischen Marktplatzes existieren in der Fachliteratur zudem weitere Begriffe, die teilweise als synonym anzusehen sind.⁵⁰¹

⁴⁹⁴ Vgl. KOLLMANN (2016), S. 459.

⁴⁹⁵ Vgl. KOLLMANN (2016), S. 459.

⁴⁹⁶ Vgl. KOLLMANN (2016), S. 459.

⁴⁹⁷ Vgl. KOLLMANN (2016), S. 459.

⁴⁹⁸ Vgl. GRIEGER (2003), S. 280.

⁴⁹⁹ Vgl. KOLLMANN (2016), S. 457.

⁵⁰⁰ Vgl. GRIEGER (2003), S. 281.

⁵⁰¹ Als Beispiele sind „B2B marketplace“, „e-hub“ und „E-Marketplace“ zu nennen. Vgl. KOLLMANN (2016), S. 457, sowie GRIEGER (2003), S. 282.

5.1.3 Merkmale von elektronischen Marktplätzen

Elektronische Marktplätze bringen mehrere Käufer und Verkäufer virtuell an einem zentralen Ort zusammen.⁵⁰² Sie sind 24 Stunden am Tag verfügbar.⁵⁰³ Auch sind sie kompetitiver als traditionelle Marktplätze und können die Transaktionskosten der Teilnehmer reduzieren.⁵⁰⁴

GRIEGER identifiziert für elektronische Marktplätze zwei typische Aspekte: Sie bieten ihren Teilnehmern einen institutionellen und sozialen Kontext.⁵⁰⁵

- Im *institutionellen Kontext* wird der elektronische Marktplatz von seinen Teilnehmern als Medium gesehen, welches ihnen Rollen⁵⁰⁶ zuweist und Transaktionen von Informationen, Sachgütern, Dienstleistungen sowie Zahlungen unterstützt.⁵⁰⁷ Darüber hinaus stellt der elektronische Marktplatz eine Infrastruktur zur Verfügung, welche mit Hilfe von festgelegten Prozessen, Protokollen und einer gemeinsamen Sprache die Rahmenbedingungen der Interaktionen innerhalb des Marktplatzes bestimmt.⁵⁰⁸
- Im *sozialen Kontext* ist der elektronische Marktplatz als Gemeinschaft aus Akteuren zu verstehen. Diese Gemeinschaft besitzt einen bestimmten Zustand, der sich aus Attributen ihrer Mitglieder, wie bspw. vertraglichen Verpflichtungen oder Gütern, ableiten lässt.⁵⁰⁹ Die Gemeinschaft weist ihren Mitgliedern Rechte und Pflichten zu, welche diese dazu nutzen, um für sie wünschenswerte Veränderungen mit Hilfe von Transaktionen und Kommunikation herbeizuführen.⁵¹⁰

Das Produkt eines elektronischen Marktplatzes ist die von ihm bereitgestellte Vermittlungsleistung zwischen Angebot und Nachfrage.⁵¹¹ Zur Erbringung dieser Dienstleistung ist der Marktplatzbetreiber auf Informationen über die Teilnehmer des elektronischen Marktplatzes angewiesen.⁵¹²

⁵⁰² Vgl. GRIEGER (2003), S. 281 f.

⁵⁰³ Vgl. GRIEGER (2003), S. 281 f.

⁵⁰⁴ Vgl. GRIEGER (2003), S. 281.

⁵⁰⁵ Vgl. GRIEGER (2003), S. 283.

⁵⁰⁶ Gemeint sind hier primär die Rollen Käufer und Verkäufer. Jedoch sind auch Rollenzuweisungen im Rahmen intermediärer Sekundärdienstleistungen wie der Zahlungsabwicklung denkbar.
Vgl. GRIEGER (2003), S. 283.

⁵⁰⁷ Vgl. GRIEGER (2003), S. 283.

⁵⁰⁸ Vgl. GRIEGER (2003), S. 283.

⁵⁰⁹ Vgl. GRIEGER (2003), S. 283.

⁵¹⁰ Vgl. GRIEGER (2003), S. 283.

⁵¹¹ Vgl. KOLLMANN (2016), S. 529.

⁵¹² Vgl. KOLLMANN (2016), S. 529 f.

5.1.4 Arten von elektronischen Marktplätzen

5.1.4.1 Vertikale und horizontale Marktplätze

In Abhängigkeit von der elektronischen Vermittlungs- bzw. Koordinationsleistung wird in der Fachliteratur in horizontale und vertikale Marktplätze unterschieden.⁵¹³

- *Vertikale Marktplätze* bündeln Angebot oder Nachfrage für eine bestimmte Branche, weshalb sie eine detaillierte Kenntnis der jeweiligen Branche erfordern.⁵¹⁴ Sie zielen auf eine Abdeckung der gesamten Wertschöpfungskette mit elektronischen Dienstleistungen, bedienen jedoch nur einen geschlossenen, spezialisierten Teilnehmerkreis.⁵¹⁵ Sämtliche Funktionalitäten sind auf diesen Teilnehmerkreis zugeschnitten.⁵¹⁶ Es erfolgt eine brancheninterne, nach bekannten Regeln ablaufende Zusammenführung von Verkäufern und Käufern.⁵¹⁷ Vertikale Marktplätze entstehen in stark fragmentierten Branchen.⁵¹⁸
- *Horizontale Marktplätze* werden auch als „funktionale Marktplätze“ bezeichnet, da sie branchenübergreifend agieren, um bestimmte Funktionen innerhalb von Organisationen zu bedienen, wie bspw. die Beschaffung oder das Personalwesen.⁵¹⁹ Auch eine Spezialisierung auf bestimmte Produktgruppen ist denkbar, wie bspw. Büromaterial oder Computerhardware.⁵²⁰ Horizontale Marktplätze zielen auf eine bestimmte Stufe der Wertschöpfungskette, an der sie Teilnehmer aus vielen unterschiedlichen Branchen zusammenbringen können.⁵²¹

Während die Vermittlungsleistung auf vertikalen Marktplätzen in die Tiefe geht, richtet sie sich auf horizontalen Marktplätzen eher in die Breite.⁵²²

5.1.4.2 Käufer- und verkäuferorientierte sowie neutrale Marktplätze

Aus der Stakeholder-Perspektive werden elektronische Marktplätze je nach Betreiber in käufer- und verkäuferorientierte sowie neutrale Marktplätze unterschieden.⁵²³

⁵¹³ Vgl. KOLLMANN (2016), S. 472, sowie GRIEGER (2003), S. 287.

⁵¹⁴ Vgl. KOLLMANN (2016), S. 472, sowie GRIEGER (2003), S. 287.

⁵¹⁵ Vgl. KOLLMANN (2016), S. 472.

⁵¹⁶ Vgl. KOLLMANN (2016), S. 472.

⁵¹⁷ Vgl. KOLLMANN (2016), S. 472.

⁵¹⁸ Vgl. KOLLMANN (2016), S. 472.

⁵¹⁹ Vgl. GRIEGER (2003), S. 288.

⁵²⁰ Vgl. KOLLMANN (2016), S. 473.

⁵²¹ Vgl. KOLLMANN (2016), S. 472 f.

⁵²² Vgl. KOLLMANN (2016), S. 473.

⁵²³ Vgl. GRIEGER (2003), S. 287.

- Im Rahmen des „Nachfrager-Modells“⁵²⁴ entstehen *käuferorientierte Marktplätze*⁵²⁵, welche von einem oder mehreren Käufern initiiert werden.⁵²⁶ Sie versuchen möglichst viele Käufer eines Sachguts oder einer Dienstleistung zusammenzubringen, um die daraus resultierende Kundenmacht zur Reduktion von Beschaffungs- und Verwaltungskosten zu nutzen.⁵²⁷ Dies kann gelingen, da bestehende Informationsasymmetrien und die Verringerung der Suchkosten zwei Motive von Teilnehmern elektronischer Marktplätze sind.⁵²⁸ Anschließend können die zusammengefassten Sachgüter und Dienstleistungen gebündelt an individuelle Käufer, Käufernetzwerke und andere Marktplätze verkauft werden.⁵²⁹ In der Regel werden Käufer versuchen, einen preisorientierten Marktplatz zu schaffen, um für sich und andere Käufer eine zuvor nicht gegebene Markttransparenz zu schaffen, einen Preiswettbewerb unter den Verkäufern zu initiieren und so langfristig die eigenen Beschaffungskosten zu senken.⁵³⁰
- Im Rahmen des „Anbieter-Modells“⁵³¹ entstehen *verkäuferorientierte Marktplätze*⁵³², welche von einem oder mehreren Verkäufern initiiert werden.⁵³³ Sie versuchen die Angebote möglichst vieler Verkäufer in einem zentralen Katalog zusammenzufassen und diesen Verkäufern so ein Forum zur Präsentation ihrer Produkte zu geben.⁵³⁴ Sie entstehen vor allem in Bereichen mit einer relativ hohen Konzentration und einer hohen Marktmacht seitens der Verkäufer.⁵³⁵ Teilweise versuchen Verkäufer durch die Etablierung eines eigenen elektronischen Marktplatzes die Produktinformation und -differenzierung in den Mittelpunkt zu stellen und so Kunden von rein preisorientierten Marktplätzen abzuwerben.⁵³⁶

⁵²⁴ Vgl. KOLLMANN (2016), S. 477.

⁵²⁵ In der Fachliteratur auch als „nachfrageseitige Marktplätze“ bezeichnet. Vgl. SCHWICKERT/PFEIFFER (2000), S. 20.

⁵²⁶ Vgl. KOLLMANN (2016), S. 477.

⁵²⁷ Vgl. GRIEGER (2003), S. 287.

⁵²⁸ Vgl. KOLLMANN (2016), S. 476.

⁵²⁹ Vgl. GRIEGER (2003), S. 287.

⁵³⁰ Vgl. KOLLMANN (2016), S. 477 f.

⁵³¹ Vgl. KOLLMANN (2016), S. 476.

⁵³² In der Fachliteratur auch als „angebotsseitige Marktplätze“ bezeichnet. Vgl. SCHWICKERT/PFEIFFER (2000), S. 19.

⁵³³ Vgl. KOLLMANN (2016), S. 476.

⁵³⁴ Vgl. KOLLMANN (2016), S. 478 f., sowie GRIEGER (2003), S. 287.

⁵³⁵ Vgl. KOLLMANN (2016), S. 476.

⁵³⁶ Vgl. KOLLMANN (2016), S. 476.

- Im Rahmen des „Makler-Modells“⁵³⁷ entstehen *neutrale Marktplätze*⁵³⁸, welche von einer neutralen Partei betrieben werden, weshalb diese Marktplätze als unabhängige Handelsvermittler sowohl für Käufer als auch Verkäufer gleichermaßen attraktiv erscheinen können.⁵³⁹ Sie stehen jedoch vor dem sogenannten „Chicken and Egg“-Problem:⁵⁴⁰ Käufer wollen nicht an dem Marktplatz teilnehmen, solange nicht eine ausreichende Anzahl von Verkäufern in diesem Marktplatz Sachgüter und Dienstleistungen anbieten, und Verkäufer wollen nicht teilnehmen, bevor sich nicht eine ausreichende Anzahl von Käufern für ihre Angebote interessiert.⁵⁴¹ Für einen elektronischen Marktplatz ist es entscheidend, die sogenannte „kritische Masse“ zu erreichen, also jene Teilnehmerzahl, ab der ein Marktplatz sowohl für Käufer als auch für Verkäufer attraktiv erscheint.⁵⁴² Da jedoch neben einer bestimmten Teilnehmeranzahl auch ein bestimmtes Verhältnis von Verkäufern zu Käufern erreicht werden muss, bevor der elektronische Marktplatz für beide Marktseiten gleichermaßen attraktiv wird, spricht KOLLMANN hier von der „doppelten kritischen Masse“.⁵⁴³

Im Gegensatz zu den Betreibern käufer- und verkäuferorientierter Marktplätze, welche selbst am Marktgeschehen ihrer Teilnehmer partizipieren, verfügen die Betreiber neutraler Marktplätze nicht über die gleichen Einsichten in die jeweilige Branche.⁵⁴⁴ Zudem sind die Verkäufer einer Branche in der Lage, das Vorhaben der Etablierung eines neutralen Marktplatzes zu boykottieren, indem sie sich entscheiden, keine Sachgüter oder Dienstleistungen auf dem neutralen Marktplatz anzubieten.⁵⁴⁵ Vor dem Hintergrund der so offerierten Nutzung eines zusätzlichen Vertriebskanals und der damit einhergehenden Kosteneinsparung und Umsatzsteigerung erscheint eine solche Blockade jedoch nicht grundsätzlich wahrscheinlich.⁵⁴⁶ Neutrale Marktplätze entstehen vor allem in Bereichen mit polypolistischer Marktstruktur.⁵⁴⁷

Bei neutralen elektronischen Marktplätzen kann zudem zwischen solchen mit aktivem zentralen Betreiber und ohne aktiven zentralen Betreiber unterschieden werden. Ein aktiver zentraler Betreiber greift aktiv in das Marktgeschehen ein, während bei einem Marktplatz ohne aktiven zentralen Betreiber lediglich ein (virtueller) Handels-

⁵³⁷ Vgl. KOLLMANN (2016), S. 476.

⁵³⁸ In der Fachliteratur auch als „Makler-Modell eines Marktplatzes“, „maklerseitige Marktplätze“ oder „vermittlerseitige Marktplätze“ bezeichnet. Vgl. KOLLMANN (2016), S. 478 ff.

⁵³⁹ Vgl. KOLLMANN (2016), S. 478, sowie GRIEGER (2003), S. 287.

⁵⁴⁰ Vgl. KOLLMANN (2016), S. 496, sowie GRIEGER (2003), S. 287.

⁵⁴¹ Vgl. KOLLMANN (2016), S. 496, sowie GRIEGER (2003), S. 287.

⁵⁴² Vgl. KOLLMANN (2016), S. 496.

⁵⁴³ Vgl. KOLLMANN (2016), S. 496 f.

⁵⁴⁴ Vgl. KOLLMANN (2016), S. 476.

⁵⁴⁵ Vgl. KOLLMANN (2016), S. 476.

⁵⁴⁶ Vgl. KOLLMANN (2016), S. 476.

⁵⁴⁷ Vgl. KOLLMANN (2016), S. 478.

raum zur Verfügung gestellt wird, also eine Plattform, auf der beliebige Verkäufer ihre Sachgüter und Dienstleistungen anbieten können.⁵⁴⁸

Für die beiden Marktseiten – Käufer und Verkäufer – bestehen im Falle der Nutzung eines neutralen Marktplatzes die gleichen Anreize, die bereits zur Gründung eines eigenen Marktplatzes aufgeführt wurden: So erschließen sich die Verkäufer durch ihre Teilnahme einen zusätzlichen Vertriebskanal und die Käufer profitieren von der gesteigerten Markttransparenz.⁵⁴⁹ Zudem verfügt der Marktplatzbetreiber über vollständige Informationen über seinen Markt, die Teilnehmer sowie die Angebote und Nachfragen durch deren unabhängige Analyse, Strukturierung und Bewertung er einen weiteren Mehrwert für beide Marktseiten schafft.⁵⁵⁰ Er kann den Teilnehmern somit eine unabhängige Intermediationsfunktion offerieren, wenn er ihnen seine Neutralität, Unabhängigkeit und Glaubwürdigkeit versichern kann.⁵⁵¹

Neutrale Marktplätze entstehen vor allem in Bereichen mit einer starken Fragmentierung, in Nischenmärkten sowie in Märkten mit großen Informationsasymmetrien zwischen Käufern und Verkäufern.⁵⁵² In Märkten mit großen Informationsasymmetrien werden elektronische Marktplätze fast ausschließlich von unabhängigen Intermediären initiiert, da zum einen die Verkäuferseite kein Interesse an einer höheren Markttransparenz hat, welche den Wettbewerb fördern würde, und zum anderen die Käuferseite nicht über genügend Ressourcen verfügt, um eigene Marktplätze zur Nachfragebündelung zu etablieren.⁵⁵³

Abbildung 7 fasst die verschiedenen Kategorisierungen von Marktplätzen zusammen.

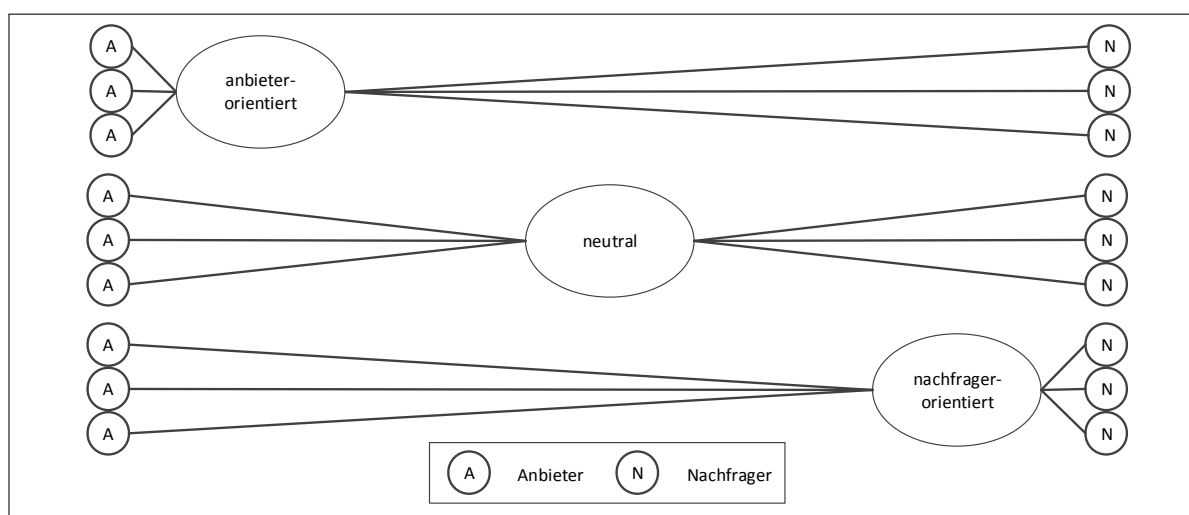


Abbildung 7: Arten elektronischer Marktplätze aus Stakeholder-Perspektive⁵⁵⁴

⁵⁴⁸ Vgl. KOLLMANN (2016), S. 478 f.

⁵⁴⁹ Vgl. KOLLMANN (2016), S. 479 f.

⁵⁵⁰ Vgl. KOLLMANN (2016), S. 479 f.

⁵⁵¹ Vgl. KOLLMANN (2016), S. 479 f.

⁵⁵² Vgl. KOLLMANN (2016), S. 481.

⁵⁵³ Vgl. KOLLMANN (2016), S. 481 f.

5.1.4.3 Offene und geschlossene Marktplätze

Es werden offene und geschlossene Marktplätze unterschieden, abhängig davon, ob der Zugang zu ihnen jedem Interessenten oder nur einer bestimmten Zielgruppe offensteht.⁵⁵⁵

- *Offene Marktplätze* sind jedem Interessenten zugänglich und die Zahl der Teilnehmer ist nicht beschränkt.⁵⁵⁶ Die Teilnehmer sind sich häufig unbekannt, was Sicherheits- und Authentizitätsprüfungen erfordert.⁵⁵⁷ Es herrscht ein geringer Grad an Kollaboration und Informationsaustausch.⁵⁵⁸
- *Geschlossene Marktplätze* sind nur einem beschränkten Teilnehmerkreis zugänglich, wodurch ihre Teilnehmerzahlen gering ausfallen.⁵⁵⁹ Die Teilnehmer sind miteinander bekannt und untereinander vernetzt, was regelmäßig Sicherheits- und Authentizitätsprüfungen ersetzt.⁵⁶⁰ Es herrscht ein hoher Grad an Kollaboration und Informationsaustausch.⁵⁶¹

5.1.4.4 Marktplätze mit und ohne Preismechanismus

Aus preisorientierter Sicht lassen sich elektronische Marktplätze in solche mit fixen, mit dynamischen und ohne Preismechanismen einteilen.⁵⁶²

- Sind die Preise innerhalb des Marktplatzes fix, so liegt ein *katalogbasiertes System* vor.⁵⁶³ Katalogbasierte Systeme digitalisieren die Kataloge mehrere Verkäufer, ermöglichen einen breiten Informationsaustausch und die Möglichkeit zur Kollaboration.⁵⁶⁴ Sie sind geeignet für Märkte mit hoher Fragmentierung, da sie den Käufern das bequeme Einkaufen an einem zentralen Punkt im Internet ermöglichen, die Markttransparenz erhöhen sowie Vergleichsmöglichkeiten eröffnen.⁵⁶⁵ Zudem bieten sie prinzipiell die Möglichkeit der Nachfragebündelung.⁵⁶⁶

⁵⁵⁴ Quelle: eigene Darstellung in Anlehnung an KOLLMANN (2016), S. 475.

⁵⁵⁵ Vgl. GRIEGER (2003), S. 287.

⁵⁵⁶ Vgl. GRIEGER (2003), S. 289.

⁵⁵⁷ Vgl. GRIEGER (2003), S. 289.

⁵⁵⁸ Vgl. GRIEGER (2003), S. 289.

⁵⁵⁹ Vgl. GRIEGER (2003), S. 289.

⁵⁶⁰ Vgl. GRIEGER (2003), S. 289.

⁵⁶¹ Vgl. GRIEGER (2003), S. 289.

⁵⁶² Vgl. GRIEGER (2003), S. 287.

⁵⁶³ Vgl. GRIEGER (2003), S. 288, sowie KOLLMANN (2001), S. 86.

⁵⁶⁴ Vgl. GRIEGER (2003), S. 288.

⁵⁶⁵ Vgl. GRIEGER (2003), S. 288, sowie KOLLMANN (2001), S. 86.

⁵⁶⁶ Vgl. KOLLMANN (2001), S. 86.

- Ist der Preismechanismus dynamisch, so liegt ein *auktionsbasiertes System* oder ein Börsensystem vor.⁵⁶⁷ Jede Art dynamischer Preisbildung ist auf eine Preisreduktion aus und transaktionsorientiert.⁵⁶⁸
- *Schwarze Bretter* oder *Portale* bieten ihren Teilnehmern eine Plattform in einem kommerziellen Umfeld, die eigentlichen Preisverhandlungen finden jedoch außerhalb der Plattform statt.⁵⁶⁹

5.1.5 Anforderungen an elektronische Marktplätze

Ein elektronischer Marktplatz muss technischen Anforderungen genügen, um seine Koordinationsfunktion erfüllen zu können.⁵⁷⁰ Die dem Marktplatz zugrundeliegende Software muss die von den teilnehmenden Verkäufern benötigten Handelskomponenten (bspw. Einstellung und Verwaltung von Produkten, Kontaktmanagement, Kommunikationstools und Auftragsbestätigung) bereitstellen.⁵⁷¹ Darüber hinaus existieren Anforderungen in den Bereichen Sicherheit, Zahlungsabwicklung und evtl. Logistik.⁵⁷²

Die einem elektronischen Marktplatz zugrundeliegende Software sollte den allgemeinen Qualitätsmerkmalen internetbasierter Software entsprechen.⁵⁷³ Dies betrifft laut KOLLMANN insbesondere die unterbrechungsfreie Erreichbarkeit, da eine Unterbrechung der Erreichbarkeit in der Endphase von Auktionen dazu führt, dass nicht alle Käufer oder Verkäufer ihre Gebote abgeben können.⁵⁷⁴ Weitere Qualitätsmerkmale ergeben sich im Bereich der Sicherheit: Aufgrund der sensiblen Natur der über einen elektronischen Marktplatz ausgetauschten Informationen, wie bspw. Produkt- oder Budgetinformationen, muss der Marktplatzbetreiber sicherstellen, dass jeder Teilnehmer nur Zugriff auf jene Daten erhält, die von ihren Absendern für ihn bestimmt wurden.⁵⁷⁵ Möglichkeiten hierzu bieten Sicherheitszertifikate, SSL-Verschlüsselung, Passwortschutz und Firewalls.⁵⁷⁶

Seitens der Marktteilnehmer stellen auch die intuitive Benutzerführung und leichte Bedienbarkeit des elektronischen Marktplatzes eine Anforderung dar⁵⁷⁷, da die Sicherstel-

⁵⁶⁷ Vgl. GRIEGER (2003), S. 288.

⁵⁶⁸ Vgl. GRIEGER (2003), S. 288.

⁵⁶⁹ Vgl. GRIEGER (2003), S. 288.

⁵⁷⁰ Vgl. KOLLMANN (2016), S. 460.

⁵⁷¹ Vgl. KOLLMANN (2016), S. 460 f.

⁵⁷² Vgl. KOLLMANN (2016), S. 461.

⁵⁷³ Vgl. KOLLMANN (2016), S. 461.

⁵⁷⁴ Vgl. KOLLMANN (2016), S. 461.

⁵⁷⁵ Vgl. KOLLMANN (2016), S. 461 f.

⁵⁷⁶ Vgl. KOLLMANN (2016), S. 461 f.

⁵⁷⁷ Vgl. KOLLMANN (2001), S. 111 f.

lung einfacher Kommunikation die schnelle Adaption des Marktplatzes fördern kann.⁵⁷⁸ Vor diesem Hintergrund spielt der Aufbau der Webseite des elektronischen Marktplatzes eine entscheidende Rolle, da sie gewährleisten muss, dass sich die Teilnehmer schnell orientieren können und einen einfachen Zugang zu den Funktionalitäten des Marktplatzes finden.⁵⁷⁹ Andererseits muss die Webseite jedoch mit Verkäufern und Käufern zwei sehr verschiedenen, sich oftmals nicht überschneidenden Anforderungsgruppen gerecht werden: Verkäufer interessieren sich für das Einstellen von Angeboten, Käufer für das Auffinden von Angeboten.⁵⁸⁰ Nach der Anmeldung sollte die Webseite des elektronischen Marktplatzes seinen Teilnehmern eine differenzierte Ansicht zur Verfügung stellen, je nach dem, ob sie Käufer oder Verkäufer sind.⁵⁸¹

Elektronische Marktplätze müssen, um ihre Vorteile gegenüber realen Marktplätzen nutzen zu können, die bestehenden Kommunikationssysteme der Marktteilnehmer in ihren Koordinationsprozess integrieren. Die Integrationsfähigkeit stellt somit eine weitere technische Anforderung an einen elektronischen Marktplatz dar.⁵⁸²

KOLLMANN unterscheidet drei Bereiche der Integrationsfähigkeit einer Software für elektronische Marktplätze:

1. **systemseitige Integration:** Die Software muss eine Anbindung von ERP-Systemen, Warenwirtschaftssystemen und externen Katalogsystemen unterstützen.⁵⁸³ Da die Systeme von Käufern und Verkäufern gleichermaßen integriert werden müssen, steigt die Komplexität dieser Anforderung, da nicht alle Teilnehmersysteme über die gleichen Schnittstellen verfügen.⁵⁸⁴
2. **prozessbezogene Integration:** Die Software muss bspw. bei der Übermittlung von Bestellungen, Lieferscheinen und Rechnungen eine unternehmensübergreifende Verknüpfung von Prozessen ermöglichen.⁵⁸⁵
3. **informationstechnische Integration:** Die Unterstützung unterschiedlicher Formate und Protokolle muss gewährleistet sein, da der Marktplatzbetreiber bestehende Formate und Prozesse unterstützen muss, anstatt die Unterstützung eigener Formate und Prozesse zu fordern.⁵⁸⁶

⁵⁷⁸ Vgl. KOLLMANN (2016), S. 462.

⁵⁷⁹ Vgl. KOLLMANN (2016), S. 462.

⁵⁸⁰ Vgl. KOLLMANN (2016), S. 462.

⁵⁸¹ Vgl. KOLLMANN (2016), S. 462.

⁵⁸² Vgl. KOLLMANN (2016), S. 462.

⁵⁸³ Vgl. KOLLMANN (2016), S. 463.

⁵⁸⁴ Vgl. KOLLMANN (2016), S. 463.

⁵⁸⁵ Vgl. KOLLMANN (2016), S. 463.

⁵⁸⁶ Vgl. KOLLMANN (2016), S. 463.

Des Weiteren sieht sich ein elektronischer Marktplatz, wenn er als Katalogsystem organisiert ist, mit den Problemen der Produktklassifikation⁵⁸⁷, des Katalogaustauschs⁵⁸⁸ sowie des Katalogmanagements⁵⁸⁹ konfrontiert.

KOLLMANN stellt fest, dass es im Rahmen der Organisation von Informationen zu einer *Koordinationslücke* kommen kann.⁵⁹⁰ Eine Koordinationslücke beschreibt die durch Reibungsverluste zwischen Käufer und Verkäufer entstehenden Ineffizienzen der marktlichen Abstimmungsprozesse.⁵⁹¹ Als Beispiel wird angeführt, dass herkömmliche Suchmaschinen bei der Suche nach einem gebrauchten Automobil viele Tausend Suchergebnisse präsentieren, jedoch nicht effizient helfen, dieser Masse an digitalen Handelsinformationen Herr zu werden.⁵⁹² Es ist eine prozessbezogene Anforderung an einen elektronischen Marktplatz, die von ihm verarbeiteten Informationen auf bestimmte Themenfelder zu konzentrieren und auf relevante Inhalte zu reduzieren.⁵⁹³

Die Handelsprozesse innerhalb des elektronischen Marktplatzes lassen sich nach dem *Transaktionsphasenmodell* differenzieren. Es bestehen hierbei verschiedene Abgrenzungen für die Unterteilung des elektronischen Transaktionsprozesses: So kann die elektronische Transaktion in die vier Phasen Information, Vereinbarung, Abwicklung und After-Sales unterteilt werden.⁵⁹⁴ Jedoch ist auch eine Unterteilung in fünf Phasen denkbar, wie REBSTOCK zeigt: Hier wird die Transaktion in die Phasen Wissen, Absicht, Vereinbarung, Abwicklung und Betreuung unterteilt.⁵⁹⁵ REBSTOCK stellt weiterhin fest, dass die Unterteilung des elektronischen Transaktionsprozesses sich zumeist nur begrifflich unterscheidet und in der Regel inhaltlich identisch ist.⁵⁹⁶

Zur genaueren Einordnung werden im Rahmen dieser Forschungsarbeit folgende drei Phasen definiert:

- **Informationsphase:** In dieser Phase wird der Kontakt zwischen Käufer und Verkäufer hergestellt.⁵⁹⁷ Es werden Informationen zur möglichen Transaktion ausge-

⁵⁸⁷ Probleme bei der Produktklassifikation bestehen vor allem in unterschiedlichen Zuordnungen eines Produkts durch verschiedene Verkäufer. Vgl. KOLLMANN (2016), S. 464 f.

⁵⁸⁸ Probleme beim Austausch von Katalogdaten bestehen vor allem in proprietären Formaten der Verkäufer. Da diese jedoch ein Interesse an der möglichst hohen Verbreitung ihrer Angebote haben, wurden Katalogaustaschformate entwickelt. Vgl. KOLLMANN (2016), S. 466 ff.

⁵⁸⁹ Probleme beim Katalogmanagement entstehen vor allem durch die unstetige Qualität der Kataloginhalte, welche aus fehlerhaften Datensätzen oder heterogenen Datenquellen und -formaten resultieren kann. Vgl. KOLLMANN (2016), S. 466 f.

⁵⁹⁰ Vgl. KOLLMANN (2016), S. 491.

⁵⁹¹ Vgl. KOLLMANN (2016), S. 491.

⁵⁹² Vgl. KOLLMANN (2016), S. 491.

⁵⁹³ Vgl. KOLLMANN (2016), S. 491.

⁵⁹⁴ Vgl. KOLLMANN (2016), S. 501 f.

⁵⁹⁵ Vgl. REBSTOCK (2000), S. 5.

⁵⁹⁶ Vgl. REBSTOCK (2000), S. 9.

⁵⁹⁷ Vgl. KOLLMANN (2016), S. 501 f., sowie REBSTOCK (2000), S. 8.

tauscht, wie bspw. Preise, Produktspezifikationen und rechtliche Rahmenbedingungen.⁵⁹⁸

- **Vereinbarungsphase:** In der Vereinbarungphase wird von Käufer und Verkäufer zunächst jeweils eine Transaktionsabsicht signalisiert.⁵⁹⁹ Im Anschluss finden die Verhandlungen zu den Konditionen der möglichen Transaktion statt, wie bspw. zu Mengen, Preisen und Lieferkonditionen.⁶⁰⁰ Die Ergebnisse dieser Verhandlungen sind rechtsverbindlich.⁶⁰¹
- **Abwicklungsphase:** Kommt in der Vereinbarungphase eine Vereinbarung über eine Transaktion zustande, so wird diese in der Abwicklungsphase durchgeführt, d.h. die Transaktionspartner erbringen gegenseitig die von ihnen beschriebenen Leistungen.⁶⁰²

Mit Abschluss der Abwicklungsphase kann die geschäftliche Beziehung der Transaktionspartner enden.⁶⁰³ Es ist jedoch auch denkbar, dass eine Kundenbetreuung über den Kauf hinaus stattfindet, da eine fehlende Kundenbetreuung durchaus einen Wettbewerbsnachteil für Unternehmen darstellen kann.⁶⁰⁴ In dieser Nachkaufphase lassen sich neben Umtausch, Preisminderung und Nachbesserung auch gewinnbringende Zusatzleistungen vermitteln, wie bspw. die Wartung des verkauften Gutes.⁶⁰⁵

5.1.6 Voraussetzungen für den Aufbau von elektronischen Marktplätzen

KOLLMANN identifiziert Bedingungen, welche allgemein erfüllt sein müssen, damit in einem Markt überhaupt ein elektronischer Marktplatz entstehen kann.⁶⁰⁶ Zunächst muss hierbei untersucht werden, ob es im anvisierten Bereich zu jenen Problemen kommt, welche die Entstehung eines elektronischen Marktplatzes begünstigen.⁶⁰⁷

Begünstigt wird die Etablierung eines Marktplatzes durch ...

- einen hochfragmentierten Bereich mit niedriger Konzentration,
- niedrige Markttransparenz,

⁵⁹⁸ Vgl. KOLLMANN (2016), S. 501 f.

⁵⁹⁹ Vgl. KOLLMANN (2016), S. 502, sowie REBSTOCK (2000), S. 8.

⁶⁰⁰ Vgl. KOLLMANN (2016), S. 501 f., sowie REBSTOCK (2000), S. 8.

⁶⁰¹ Vgl. KOLLMANN (2016), S. 501.

⁶⁰² Vgl. KOLLMANN (2016), S. 502, sowie REBSTOCK (2000), S. 8.

⁶⁰³ Vgl. KOLLMANN (2016), S. 502, sowie REBSTOCK (2000), S. 8.

⁶⁰⁴ Vgl. KOLLMANN (2016), S. 502, sowie REBSTOCK (2000), S. 8.

⁶⁰⁵ Vgl. KOLLMANN (2016), S. 502, sowie REBSTOCK (2000), S. 8.

⁶⁰⁶ Vgl. KOLLMANN (2016), S. 585 f.

⁶⁰⁷ Vgl. KOLLMANN (2016), S. 585 f.

- große Informationsasymmetrien,
- eine hohe geographische Fragmentierung der Zielgruppe, welche lokale Wissensschränken begünstigt,
- Prozessineffizienzen bezüglich des Bestellwesens sowie der Informations- und Preisfindung sowie
- gut elektronisch beschreibbare, standardisierte Sachgüter und Dienstleistungen.⁶⁰⁸

Zur weiteren Einordnung der Erfolgsaussichten eines elektronischen Marktplatzes sind der zu erwartende Teilnehmerkreis, das Wettbewerbsumfeld und das Profitpotential zu betrachten.⁶⁰⁹

Existieren im anvisierten Bereich bereits andere elektronische Marktplätze oder reale Intermediäre, so können diese je nach ihrer Positionierung und Liquidität genauso eine Rolle für die Erfolgchancen eines elektronischen Marktplatzes spielen wie die Marktakzeptanz neuer Intermediäre.⁶¹⁰ Sind die potentiellen Teilnehmer internet- oder zumindest technologieaffin und ihre Geschäftsprozesse bereits elektronisch gestützt, so steigt die Wahrscheinlichkeit, dass sie einen elektronischen Marktplatz zur Abwicklung ihrer Transaktionen akzeptieren werden.⁶¹¹ Sind bei den potentiellen Teilnehmern jedoch nur eine geringe E-Commerce-Erfahrung⁶¹², niedrige Handelsvolumina, eine niedrige zu erwartende Datenqualität oder eine mangelnde technische Infrastruktur festzustellen, so sinkt die Erfolgswahrscheinlichkeit der Etablierung eines elektronischen Marktplatzes.⁶¹³

5.1.7 Geschäftsmodelle

Unabhängig von ihrer Art müssen elektronische Marktplätze zur Finanzierung ihres Betriebs Umsätze generieren. Zu diesem Zweck müssen vom Marktplatzbetreiber im Rahmen eines Geschäftsmodells verschiedene Umsatzquellen auf ihre Tauglichkeit geprüft werden.

Die möglichen Umsatzquellen sind:

- **Grundgebühr:** Für den Zugang zum Marktplatz ist eine feste monatliche oder jährliche Gebühr zu entrichten.⁶¹⁴

⁶⁰⁸ Vgl. KOLLMANN (2016), S. 585 f.

⁶⁰⁹ Vgl. KOLLMANN (2016), S. 587.

⁶¹⁰ Vgl. KOLLMANN (2016), S. 587.

⁶¹¹ Vgl. KOLLMANN (2016), S. 586.

⁶¹² Unter „E-Commerce“ wird der elektronische Verkauf verstanden, d.h. eine Unterstützung von Kommunikations- und Geschäftsprozessen im Absatzbereich mit Hilfe von elektronischen Kommunikationstechnologien. Vgl. REBSTOCK (2000), S. 6 f.

⁶¹³ Vgl. KOLLMANN (2016), S. 586 f.

⁶¹⁴ Vgl. KOLLMANN (2001), S. 127.

- **Nutzungsgebühren:** Für den Abruf bestimmter Informationen, wie bspw. durch die Nutzung von Branchendiensten, ist eine Gebühr zu entrichten.⁶¹⁵
- **Zusatzleistungen:** Für Spezialdienste, wie die Benachrichtigung über die Verfügbarkeit bestimmter Informationen, wie bspw. die Verfügbarkeit neuer Angebote, ist eine Gebühr zu entrichten.⁶¹⁶
- **Provision:** Beim Abschluss einer Transaktion über den Marktplatz ist eine Gebühr in Form einer prozentualen Vermittlungsprovision auf den Verkaufspreis zu entrichten.⁶¹⁷
- **Nebenleistungen:** Durch die starke Fokussierung des elektronischen Marktplatzes besteht die Möglichkeit, weitere Umsatzquellen zu erschließen, bspw. durch das Schalten branchenspezifischer Werbung.⁶¹⁸

Hierbei ist zu beachten, dass sich das Geschäftsmodell (auch: Erlösmodell oder Einnahmenmodell) eines elektronischen Marktplatzes aus beliebigen Kombinationen dieser fünf Umsatzquellen zusammensetzen kann.

5.1.8 Relevanz für die wissenschaftlichen Probleme

Die hier erörterten Grundlagen zu elektronischen Marktplätzen zeigen, dass das zu konzipierende AFEX-System die Eigenschaften eines elektronischen Marktplatzes besitzt.

Es ist ein funktionaler Marktplatz, der in den verschiedenen Teilbranchen multimodaler Transporte (Schienengüterverkehr, Straßengüterverkehr etc.) eine branchenübergreifende Koordination der Transportdienstleistungsvermittlung erfüllt.⁶¹⁹

Es ist ein neutraler Marktplatz, da der Marktplatz vom Kollektiv der Agenten selbst betrieben wird.⁶²⁰ Wie beschrieben, unterliegt auch das AFEX-System dem Problem der „doppelten kritischen Masse“. ⁶²¹ Die polypolistischen Marktstrukturen im Transportwesen begünstigen die Etablierung eines neutralen Systems.

Durch die geplante dynamische Preisbildung aufgrund der Verwendung von zweiseitigen kombinatorischen Auktionen ist das AFEX-System ein auktionsbasiertes System.⁶²²

⁶¹⁵ Vgl. KOLLMANN (2001), S. 127.

⁶¹⁶ KOLLMANN bezeichnet dies als Bearbeitungsgebühr. Vgl. KOLLMANN (2001), S. 127.

⁶¹⁷ Vgl. KOLLMANN (2016), S. 65.

⁶¹⁸ Vgl. KOLLMANN (2016), S. 64.

⁶¹⁹ Vgl. Kapitel 5.1.4.1, S. 63.

⁶²⁰ Vgl. Kapitel 5.1.4.2, S. 64.

⁶²¹ Vgl. Kapitel 5.1.4.2, S. 65.

⁶²² Vgl. Kapitel 5.1.4.4, S. 68.

5.2 Multi-Agenten-Systeme

5.2.1 Einführung

Die Informatik hat zur Adressierung immer komplexer werdender Problemstellungen im Laufe der vergangenen Jahrzehnte verschiedene Paradigmen hervorgebracht. Neben wichtigen Meilensteinen, wie der Einführung objektorientierter Programmierung und der Etablierung von *design patterns* im Rahmen von Softwarearchitektur, ist das für diese Forschungsarbeit entscheidende Paradigma das agentenbasierter Softwaresysteme und der *agentenorientierten Programmierung*.

Das Konzept *intelligenter Software-Agenten* stellt ein verhältnismäßig junges Forschungsfeld der Informatik dar und hat sowohl in der Forschung zu künstlicher Intelligenz als auch in der breiteren Informatik seit den 1990er Jahren an Bedeutung gewonnen.⁶²³ Der Grund hierfür ist, dass die Komplexität von Softwaresystemen stetig zunimmt und damit die Aufgaben, welche diese Softwaresysteme erfüllen müssen, inhärent komplexer werden.⁶²⁴ Somit steigt der Bedarf an Werkzeugen, um diese komplexer werdenden Problemstellungen möglichst einfach und akkurat auf natürliche Weise zu modellieren.⁶²⁵

JENNINGS argumentiert, dass die etablierten Paradigmen der Softwareentwicklung zwar in die richtige Richtung weisen, jedoch darunter leiden, dass sie die Interaktionen zwischen verschiedenen Entitäten der Software zu rigide definieren und aufgrund unzureichender Mechanismen nicht in der Lage sind, die den Systemen inhärenten organisationalen Strukturen adäquat zu repräsentieren.⁶²⁶ JENNINGS stellt weiterhin die These auf, dass der agentenorientierte Ansatz der Softwareentwicklung in der Lage ist, diese Probleme zu lösen.⁶²⁷ Seiner Meinung nach versetzen agentenorientierte Ansätze Programmierer in die Lage, komplexe, verteilte Softwaresysteme zu entwerfen und zu implementieren. Zudem soll sich der agentenorientierte Ansatz als Paradigma der Softwareentwicklung etablieren.⁶²⁸

5.2.2 Begriff des Multi-Agenten-Systems

Multi-Agenten-Systeme werden definiert als Kollektive, welche sich aus mehreren *autonomen* Softwarekomponenten – den sog. Software-Agenten – zusammensetzen, um einem System eine intendierte Funktionalität zu verleihen.⁶²⁹

⁶²³ Vgl. SUDEIKAT (2010), S. 11, JENNINGS (2000), S. 277, sowie WOOLDRIGE/JENNINGS (1995), S. 115.

⁶²⁴ Vgl. JENNINGS (2000), S. 278.

⁶²⁵ Vgl. JENNINGS (2000), S. 278, sowie BURMEISTER et al. (1997), S. 1.

⁶²⁶ Vgl. JENNINGS (2000), S. 278.

⁶²⁷ Vgl. JENNINGS (2000), S. 278.

⁶²⁸ Vgl. JENNINGS (2000), S. 278.

⁶²⁹ Vgl. SUDEIKAT (2010), S. 11, sowie BURMEISTER et al. (1997), S. 1.

Die Begrifflichkeit des Multi-Agenten-Systems und seine charakterisierenden Merkmale sind nicht trennscharf definiert. Insbesondere bzgl. der Bedeutung und Abgrenzung von Multi-Agenten-Systemen zu mobilen Agenten-Systemen berichtet FRIEDRICH von verschiedenen Auffassungen in der Literatur.⁶³⁰ Für die Zwecke dieser Forschungsarbeit wird die obige, in Anlehnung an SUDEIKAT und BURMEISTER et al. gefasste Definition gewählt, da die Frage der Mobilität und informationstechnischen Kategorisierung der Agenten für die Ziele dieser Forschungsarbeit nicht relevant ist.⁶³¹

Eine zentrale Bedeutung, sowohl für das Konzept der Software-Agenten als auch für die Ziele dieser Forschungsarbeit, hat jedoch die Frage: Was ist ein Agent?⁶³²

5.2.3 Begriff des Agenten

Die grundlegenden Konzepte und Ideen agentenbasierter Programmierung sind nicht präzise abgegrenzt. Es können jedoch eine Reihe von Feststellungen dazu getroffen werden, welche Eigenschaften ein „Agent“ im Sinne dieser Forschungsarbeit haben sollte.⁶³³

So geben WOOLDRIGE und JENNINGS die Definition des Agenten⁶³⁴ in einer engeren und einer weiteren Fassung. Die weitere Fassung erfordert für das Vorhandensein eines Agenten lediglich die Erfüllung folgender Kriterien:

- **Autonomie:** Agenten agieren ohne permanente Steuerung durch einen Menschen und haben eine gewisse Kontrolle über ihr eigenes Handeln.⁶³⁵
- **Soziale Kompetenz:** Agenten interagieren mit anderen Agenten über eine agentenspezifische Sprache.⁶³⁶

⁶³⁰ Vgl. FRIEDRICH (2009), S. 17 f.

⁶³¹ Gemeint ist hier bspw. die Frage, ob die betrachteten Agenten ein mobiles Agenten-System oder ein Multi-Agenten-System bilden. Dies würde eine Zuordnung derselben entweder zum Forschungsbereich der Künstlichen Intelligenz oder der Systemprogrammierung zur Folge haben. Vgl. FRIEDRICH (2009), S. 17 u. 21.

⁶³² Vgl. WOOLDRIGE/JENNINGS (1995), S. 116.

⁶³³ Diese Definition wird in weiten Teilen von JENNINGS übernommen. Sie dient dazu, Entitäten, welche klar als Agenten zu bezeichnen sind, von jenen abzugrenzen, welche dies nicht sind. Eine weitergehende Differenzierung des Begriffs Agent über dieses Kapitel hinaus wird nicht vorgenommen. Die relevante Literatur bietet ausführlichere Ausführungen. Vgl. JENNINGS (2000), S. 280 ff., sowie WOOLDRIGE/JENNINGS (1995), S. 116 ff.

⁶³⁴ Die im Rahmen dieser Forschungsarbeit betrachteten Agenten sind Software-Agenten. Während Teile der Fachliteratur zu Agenten sich mit Fragestellungen befassen, welche sich nicht zwangsläufig auf Software-Agenten beziehen, ist in dieser Forschungsarbeit mit dem Begriff „Agent“ stets ein Software-Agent gemeint. Aus Gründen der Lesbarkeit wird im Folgenden der Begriff „Agent“ synonym für „Software-Agent“ verwendet.

⁶³⁵ Vgl. WOOLDRIGE/JENNINGS (1995), S. 116.

⁶³⁶ Vgl. WOOLDRIGE/JENNINGS (1995), S. 116.

- **Reaktivität:** Agenten nehmen ihre Umgebung⁶³⁷ wahr und reagieren zeitnah auf Veränderungen.⁶³⁸
- **Proaktivität:** Agenten reagieren jedoch nicht ausschließlich, sondern sind auch in der Lage zielgerichtetes Verhalten zu zeigen und die Initiative zu ergreifen.⁶³⁹

Diese Definitionen grenzen Software-Agenten nicht klar von anderen Softwarekomponenten oder anderen Formen rechnerischer und natürlicher Systeme ab, welche die genannten Kriterien ebenfalls erfüllen.⁶⁴⁰ Autonome Agenten können in lebende Entitäten (*biologische Agenten*), Roboter (*robotische Agenten*) sowie rechnerische Entitäten⁶⁴¹ (*rechnerische Agenten*⁶⁴²) unterschieden werden.⁶⁴³ Software-Agenten sind nach dieser Unterteilung den rechnerischen Agenten zuzuordnen.⁶⁴⁴ Sie unterteilen sich in *aufgabenspezifische Agenten*, Computerviren (*bösartige Agenten*) sowie *Unterhaltungsagenten*.⁶⁴⁵

Abbildung 8 fasst die verschiedenen Kategorisierungen von Agenten zusammen.

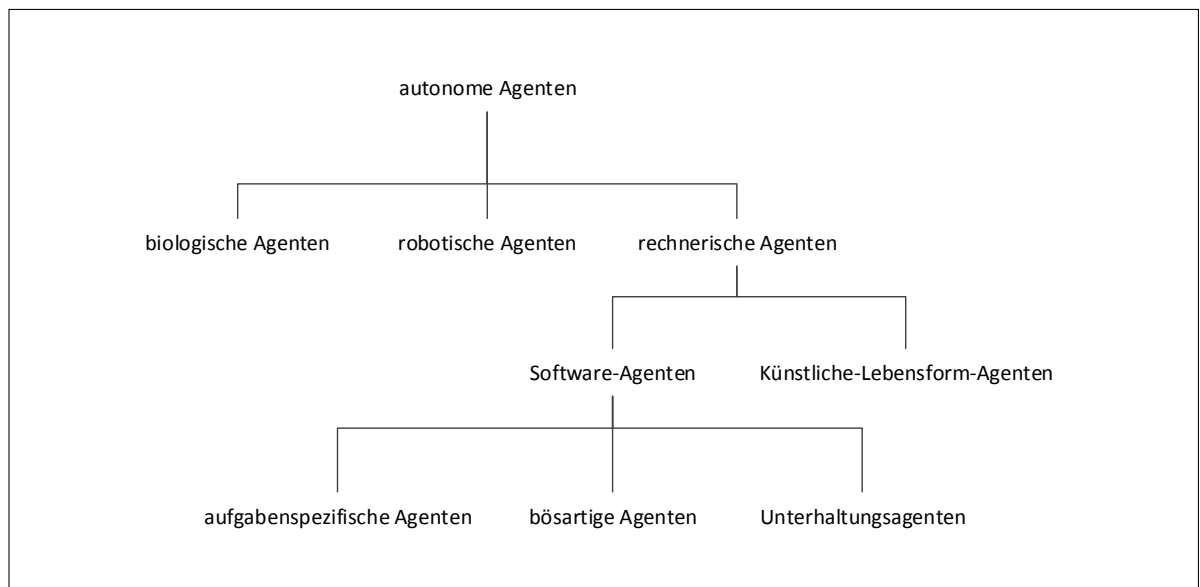


Abbildung 8: Klassifizierungen von Agenten⁶⁴⁶

⁶³⁷ Die Umgebung kann hierbei laut WOOLDRIGE und JENNINGS die reale Welt, andere Agenten, ein über eine grafische Benutzeroberfläche agierender Mensch oder das Internet sein. Vgl. WOOLDRIGE/JENNINGS (1995), S. 116.

Zusätzlich kann die Umgebung der Agenten eines Multi-Agenten-Systems selbst als Element des Multi-Agenten-Systems gesehen werden. Eine detaillierte Beschreibung von Umgebungen findet sich in Kapitel 5.2.5. Vgl. Kapitel 5.2.5, S. 83.

⁶³⁸ Vgl. WOOLDRIGE/JENNINGS (1995), S. 116.

⁶³⁹ Vgl. WOOLDRIGE/JENNINGS (1995), S. 116.

⁶⁴⁰ Vgl. SUDEIKAT (2010), S. 13.

⁶⁴¹ Im Englischen „computational entities“. Vgl. SUDEIKAT (2010), S. 13.

⁶⁴² Im Englischen „computational agents“. Vgl. SUDEIKAT (2010), S. 13.

⁶⁴³ Vgl. SUDEIKAT (2010), S. 13, sowie FRANKLIN/GRAESSER (1997), S. 27 ff.

⁶⁴⁴ Vgl. SUDEIKAT (2010), S. 13.

⁶⁴⁵ Vgl. SUDEIKAT (2010), S. 13. Diese Unterteilung von Software-Agenten ist kritisch zu hinterfragen, da bössartige und Unterhaltungsagenten ebenfalls gezielt spezifische Aufgaben verfolgen.

Aus dieser Differenzierung wird ersichtlich, dass die in dieser Forschungsarbeit behandelten Agenten eindeutig der Gruppe der aufgabenspezifischen Agenten zuzuordnen sind. Dennoch muss diese Differenzierung als Abstraktion verstanden werden, da Agenten primär ein abstraktes Hilfsmittel zur Analyse und Gestaltung von Systemen darstellen und keinen absoluten Sachverhalt; es kann keine definitive, dichotome Abgrenzung von „Agenten“ und „Nicht-Agenten“ vorgenommen werden.⁶⁴⁷ Vielmehr werden Agenten in der Informatik als Werkzeug benutzt, um Applikationsdomänen und Softwaresysteme zu konzipieren.⁶⁴⁸

In der engeren Fassung muss ein Agent – neben den zuvor genannten Kriterien – zusätzlich unter Aspekten konzipiert oder implementiert werden, welche häufiger beim Menschen Anwendung finden.⁶⁴⁹ Gemeint sind hier Charakteristika wie Wissen, Glaube, Intention und Pflicht. Einige Forscher gehen noch weiter und konzipieren *emotionale Agenten*.⁶⁵⁰

Weitere Attribute, die im Zuge der Fragestellung „Was ist ein Agent?“ diskutiert werden, sind:

- **Mobilität:** Agenten sind in der Lage sich innerhalb eines elektronischen Netzwerks zu bewegen.⁶⁵¹
- **Aufrichtigkeit:** Agenten verbreiten niemals wissentlich Fehlinformationen.⁶⁵²
- **Benevolenz:** Agenten haben keine konfliktären Ziele und werden stets tun, was von ihnen verlangt ist.⁶⁵³
- **Rationalität:** Agenten verhalten sich stets in einer Art, die der Erreichung ihrer Ziele dienlich ist.⁶⁵⁴

Agentenarchitekturen leiten die Entwicklung von Agenten an, indem sie ihre interne Struktur vorschreiben.⁶⁵⁵ Das Spektrum der möglichen Architekturen ist breit aufgestellt: Auf der einen Seite stehen *kognitive Architekturen*, welche der Entwicklung von Multi-Agenten-Systemen dienen, die sich aus intelligenten Agenten zusammensetzen.⁶⁵⁶ Diese Agenten verfügen über Wissen, welches ihnen hilft, Schlussfolgerungen bezüglich der von ihnen zu

⁶⁴⁶ Quelle: eigene Darstellung in Anlehnung an FRANKLIN/GRAESSER (1997), S. 31.

⁶⁴⁷ Vgl. SUDEIKAT (2010), S. 14.

⁶⁴⁸ Vgl. SUDEIKAT (2010), S. 14.

⁶⁴⁹ Vgl. WOOLDRIGE/JENNINGS (1995), S. 117.

⁶⁵⁰ Vgl. WOOLDRIGE/JENNINGS (1995), S. 117.

⁶⁵¹ Vgl. WOOLDRIGE/JENNINGS (1995), S. 117.

⁶⁵² Vgl. WOOLDRIGE/JENNINGS (1995), S. 117.

⁶⁵³ Vgl. WOOLDRIGE/JENNINGS (1995), S. 117.

⁶⁵⁴ Vgl. WOOLDRIGE/JENNINGS (1995), S. 117.

⁶⁵⁵ Vgl. SUDEIKAT (2010), S. 14.

⁶⁵⁶ Vgl. SUDEIKAT (2010), S. 14.

tätigenden Aktionen zu treffen.⁶⁵⁷ Auf der anderen Seite stehen *reaktive Architekturen*, innerhalb derer reaktive Agenten auf die Entwicklung ihrer Umgebung reagieren.⁶⁵⁸ Diese Agenten bilden Ereignisse nach vorgegebenen Mustern auf Aktionen oder Verhaltensweisen ab.⁶⁵⁹ Dieses Spektrum erlaubt ebenfalls *hybride Architekturen*, wodurch Agenten in die Lage versetzt werden, sowohl rein reaktives Verhalten zu zeigen als auch individuelles Verhalten zu erwägen.⁶⁶⁰

Ein Beispiel für eine Agentenarchitektur ist die „Belief - Desire - Intentions“-Architektur (BDI-Architektur).⁶⁶¹

Die Ansichten („beliefs“) eines Agenten repräsentieren die ihm zur Verfügung stehenden Informationen über seine Umgebung sowie seinen internen Zustand und determinieren, wie er sich selbst und seine Umgebung wahrnimmt.⁶⁶²

Die Wünsche („desires“) eines Agenten determinieren seine Ziele.⁶⁶³ Die Ziele eines Agenten werden in der Regel durch einen (in Bezug auf seine Umgebung oder sich selbst) nicht erreichten Zustand ausgedrückt, den der Agent herbeiführen möchte.⁶⁶⁴ Zur Erreichung seiner Ziele ist jeder Agent mit einer Reihe von ausführbaren Plänen ausgestattet.⁶⁶⁵ Pläne werden Zielen zugeordnet, um anzuzeigen, dass ein Plan eine Möglichkeit darstellt, einen bestimmten Zustand herbeizuführen bzw. ein bestimmtes Ziel oder Teilziel zu erreichen.⁶⁶⁶

Die Intentionen („intentions“) eines Agenten repräsentieren die Ziele und Pläne, welche seinem Handeln zur Erreichung eines bestimmten Zustands zugrundeliegen.⁶⁶⁷

Die Ausführung des Agenten wird von einem Interpreter gesteuert.⁶⁶⁸ Als ereignisgesteuertes Kontrollinstrument erkennt der Interpreter jene Ereignisse, welche vom Agenten verarbeitet werden müssen (bspw. eingehende Nachrichten), und stößt Aktionen an (bspw. das Ausführen von Plänen), um auf diese Ereignisse zu reagieren.⁶⁶⁹

⁶⁵⁷ Vgl. SUDEIKAT (2010), S. 14.

⁶⁵⁸ Vgl. SUDEIKAT (2010), S. 14.

⁶⁵⁹ Vgl. SUDEIKAT (2010), S. 14.

⁶⁶⁰ Vgl. SUDEIKAT (2010), S. 14.

⁶⁶¹ Vgl. SUDEIKAT (2010), S. 15.

⁶⁶² Vgl. SUDEIKAT (2010), S. 15.

⁶⁶³ Vgl. SUDEIKAT (2010), S. 15.

⁶⁶⁴ Vgl. SUDEIKAT (2010), S. 15.

⁶⁶⁵ Vgl. SUDEIKAT (2010), S. 15.

⁶⁶⁶ Vgl. SUDEIKAT (2010), S. 15.

⁶⁶⁷ Vgl. SUDEIKAT (2010), S. 14.

⁶⁶⁸ Vgl. SUDEIKAT (2010), S. 15.

⁶⁶⁹ Vgl. SUDEIKAT (2010), S. 15.

Die BDI-Architektur wendet *reaktive Planung* an: Pläne werden nicht zur Laufzeit des Agenten generiert, sondern der Agent wird bereits während der Entwicklung mit einer Reihe ausführbarer Pläne ausgestattet.⁶⁷⁰ Zur Laufzeit entscheidet der Agent, welche Pläne er ausführt.⁶⁷¹ Er legt hierzu fest, welche Ziele er erreichen möchte.⁶⁷² Für ein Ziel können hierbei zu verschiedenen Zeitpunkten ein einzelner Plan oder mehrere Pläne anwendbar sein.⁶⁷³ Im Rahmen der Zielerwägung⁶⁷⁴ entscheidet der Agent aufgrund zuvor festgelegter, konditionaler Beziehungen zwischen Zielen und Plänen, welche die Grenzen der Anwendbarkeit von Plänen definieren, welchen Plan er zur Ausführung bringt.⁶⁷⁵

Vor dem Hintergrund, dass die agentenorientierte Programmierung von Teilen der Fachliteratur als konsequente Fortführung des Paradigmas objektorientierter Programmierung gesehen wird, erstaunt es nicht, dass in der Literatur – analog zu objektorientierten Programmiersprachen – eine Vielzahl agentenorientierter Programmiersprachen entwickelt wurde.⁶⁷⁶

Es lassen sich fünf Typen agentenorientierter Sprachen identifizieren, die bei der Entwicklung von Multi-Agenten-Systemen zum Einsatz kommen:⁶⁷⁷

1. *Implementierungssprachen*⁶⁷⁸ werden eingesetzt, um ausführbare Software zu beschreiben und technisch zu realisieren.⁶⁷⁹ Mit ihnen werden grundlegende Aufgaben realisiert, wie die Generierung der Datenstrukturen, auf denen der Agent operiert.⁶⁸⁰
2. *Kommunikationssprachen*⁶⁸¹ werden eingesetzt, um die zwischen den Agenten stattfindenden nachrichtenbasierten Interaktionen zu beschreiben.⁶⁸² Diesen Sprachen kommt vor allem in heterogenen Multi-Agenten-Systemen, in denen Agenten verschiedener Architekturen zum Einsatz kommen, eine hohe Bedeutung zu, da sie die Interoperabilität der Agenten sicherstellen.⁶⁸³

⁶⁷⁰ Vgl. SUDEIKAT (2010), S. 15.

⁶⁷¹ Vgl. SUDEIKAT (2010), S. 15.

⁶⁷² Vgl. SUDEIKAT (2010), S. 15.

⁶⁷³ Vgl. SUDEIKAT (2010), S. 15.

⁶⁷⁴ Im Englischen „goal deliberation“. Vgl. SUDEIKAT (2010), S. 15.

⁶⁷⁵ Vgl. SUDEIKAT (2010), S. 15.

⁶⁷⁶ Vgl. BORDINI et al. (2006), S. 33.

⁶⁷⁷ Vgl. SUDEIKAT (2010), S. 16.

⁶⁷⁸ Im Englischen „implementation languages“. Vgl. SUDEIKAT (2010), S. 16.

⁶⁷⁹ Vgl. SUDEIKAT (2010), S. 16.

⁶⁸⁰ Vgl. SUDEIKAT (2010), S. 16.

⁶⁸¹ Im Englischen „communication languages“. Vgl. SUDEIKAT (2010), S. 16.

⁶⁸² Vgl. SUDEIKAT (2010), S. 16.

⁶⁸³ Vgl. SUDEIKAT (2010), S. 16.

3. *Verhalten beschreibende Sprachen*⁶⁸⁴ ermöglichen es, die Aktionen von Agenten innerhalb von Umgebungen zu beschreiben und hierbei von der zur Implementierung des Agenten genutzten Programmiersprache zu abstrahieren.⁶⁸⁵
4. *Wissen repräsentierende Sprachen*⁶⁸⁶ werden eingesetzt, um das den Agenten zur Verfügung stehende Wissen über sich und ihre Umgebung zu beschreiben.⁶⁸⁷ Für kognitive Agenten ermöglichen diese Beschreibungen Abwägungs- und Vorhersage-techniken.⁶⁸⁸ Für rein reaktive Agenten kann diese Sprachebene theoretisch entfallen.⁶⁸⁹ Allerdings sind die meisten Agenten nicht rein reaktive Agenten und müssen wenigstens ihre internen Informationen beschreiben, wodurch die Verwendung von Wissen repräsentierenden Sprachen in den meisten Fällen obligatorisch ist.⁶⁹⁰
5. *Spezifikationssprachen*⁶⁹¹ dienen der formalen Spezifikation der Aktionen der Agenten.⁶⁹² Sie sind hochgradig abstrahierte Meta-Sprachen und beschreiben abstrakte Konzepte von Multi-Agenten-Systemen, wie z.B. Interaktionen, Intentionen und Konditionen, denen die Agenten folgen müssen.⁶⁹³

Für die Entwicklung eines Multi-Agenten-Systems müssen diese Sprachen miteinander verbunden werden, um das System zu beschreiben.⁶⁹⁴ Die Abgrenzung der fünf Sprachtypen ist jedoch rein konzeptioneller Natur.⁶⁹⁵ Prinzipiell müssen nicht alle Sprachtypen in derselben Programmiersprache definiert sein, jedoch empfiehlt es sich im Rahmen eines Softwareentwicklungsprozesses eine Entwicklungsumgebung zu schaffen.⁶⁹⁶

Abbildung 8 fasst die verschiedenen Sprachtypen zur Programmierung von Agenten zusammen.

⁶⁸⁴ Im Englischen „behavior description languages“. Vgl. SUDEIKAT (2010), S. 16.

⁶⁸⁵ Vgl. SUDEIKAT (2010), S. 16.

⁶⁸⁶ Im Englischen „knowledge representation languages“. Vgl. SUDEIKAT (2010), S. 16.

⁶⁸⁷ Vgl. SUDEIKAT (2010), S. 16.

⁶⁸⁸ Vgl. SUDEIKAT (2010), S. 16.

⁶⁸⁹ Vgl. SUDEIKAT (2010), S. 16.

⁶⁹⁰ Vgl. SUDEIKAT (2010), S. 16.

⁶⁹¹ Im Englischen „specification languages“. Vgl. SUDEIKAT (2010), S. 16.

⁶⁹² Vgl. SUDEIKAT (2010), S. 16.

⁶⁹³ Vgl. SUDEIKAT (2010), S. 16.

⁶⁹⁴ Vgl. SUDEIKAT (2010), S. 16.

⁶⁹⁵ Vgl. SUDEIKAT (2010), S. 16.

⁶⁹⁶ Vgl. SUDEIKAT (2010), S. 16 f.

straktionsgrade und begreift Agenten als Spezialfälle von Objekten.⁷⁰³ Hierbei stellt ein Agent eine unabhängig simultan ausgeführte Software dar, ein „Objekt“, welches einen Zustand⁷⁰⁴ annimmt und mit anderen Objekten (Agenten) durch Nachrichtenaustausch kommuniziert.⁷⁰⁵ Ein derartiges Objekt ist nach zuvor definierter, weiter Fassung ein Agent.

Aus technischer Sicht sind drei Kriterien, welche Agenten von Objekten in objektorientierten Systemen unterscheiden, relevant:

1. interner/mentaler Zustand,
2. Nachrichtentypen bei der Kommunikation sowie
3. Kooperationsprotokolle.⁷⁰⁶

Interner/mentaler Zustand bedeutet, dass das Verhalten eines Agenten von seinen Ansichten, Wünschen und Intentionen⁷⁰⁷ gesteuert wird.

Nachrichtentypen bei der Kommunikation bedeutet, dass sich die Nachrichten, welche Agenten untereinander austauschen, durch ihren Typ unterscheiden.⁷⁰⁸ Alle Nachrichten haben eine inhaltsunabhängige Semantik.⁷⁰⁹

Kooperationsprotokolle können von Entwicklern genutzt werden, um den zwischen Agenten stattfindenden Dialogen einen kontextspezifischen Rahmen zu setzen.⁷¹⁰ Die Agenten können auf Basis der Protokolle auf Nachrichten reagieren und wissen, welche Nachrichten und Antworten auf gesendete Nachrichten in welchem Zustand zu erwarten sind.⁷¹¹ Kooperationsprotokolle können darüber hinaus dazu dienen, Nachrichten zwischen heterogenen Agenten – also Agenten mit unterschiedlichen internen Strukturen und Zielen – auszutauschen.⁷¹²

Die Vorteile durch die Verwendung von agentenorientierter Programmierung sind die Reduktion der Systemkomplexität aufgrund der Möglichkeiten zur präzisen und natürlichen

⁷⁰³ Vgl. BURMEISTER et al. (1997), S. 2.

⁷⁰⁴ Häufig trifft man in der Literatur auch den englischen Begriff „state“ an. Vgl. WOOLDRIGE (1997), S. 27.

⁷⁰⁵ Vgl. BURMEISTER et al. (1997), S. 2, sowie WOOLDRIGE/JENNINGS (1995), S. 117.

⁷⁰⁶ Vgl. BURMEISTER et al. (1997), S. 2.

⁷⁰⁷ Im Englischen wird diese Gruppe „beliefs, desires, intentions“ genannt. Auf diesem Konzept basiert die BDI-Architektur. Vgl. BURMEISTER et al. (1997), S. 2.

⁷⁰⁸ Vgl. BURMEISTER et al. (1997), S. 2.

⁷⁰⁹ Vgl. BURMEISTER et al. (1997), S. 2.

⁷¹⁰ Vgl. BURMEISTER et al. (1997), S. 3.

⁷¹¹ Vgl. BURMEISTER et al. (1997), S. 3.

⁷¹² Vgl. BURMEISTER et al. (1997), S. 3.

Beschreibung der Problemdomäne sowie die erhöhte Robustheit und Adaptivität der sich selbst organisierenden Agenten-Systeme.⁷¹³

5.2.5 Umgebungen

Die Umgebung wird definiert als Abstraktion der die Agenten umgebenden Verhältnisse, welche sowohl die Interaktion zwischen den Agenten als auch ihren Zugriff auf Ressourcen und Funktionalitäten vermittelt.⁷¹⁴

Die Umgebung eines Agenten beschreibt damit den Kontext, in welchem der einzelne Agent ausgeführt wird.⁷¹⁵ Typischerweise bilden alle Elemente eines Multi-Agenten-Systems, die keine Agenten sind, diese Umgebung.⁷¹⁶

Umgebungen können eine Reihe von Eigenschaften haben:⁷¹⁷

1. **zugänglich oder unzugänglich:** Umgebungen sind zugänglich, wenn Agenten in der Lage sind den Zustand der Umgebung vollständig wahrzunehmen.⁷¹⁸ Alternativ ist eine Umgebung bereits *effektiv zugänglich*, wenn ein Agent alle für seine Tätigkeit nötigen Informationen wahrnehmen kann.⁷¹⁹
2. **deterministisch oder nicht deterministisch:** Umgebungen sind deterministisch, wenn der aktuelle Zustand der Umgebung und der Agenten zukünftige Zustände vollständig festlegt.⁷²⁰ Die Beurteilung hängt jedoch von der Zugänglichkeit ab, so dass unzugängliche Umgebungen von Agenten oft als nicht deterministisch wahrgenommen werden.⁷²¹
3. **episodisch oder nicht episodisch:** Eine Episode bezeichnet die Wahrnehmung und den Aktionszyklus eines Agenten.⁷²² Wenn diese Episoden unabhängig voneinander ablaufen, also die aktuelle Episode nicht von der vorangegangenen beeinflusst wird, so nennt man die Umgebung episodisch.⁷²³

⁷¹³ Vgl. BURMEISTER et al. (1997), S. 3.

⁷¹⁴ Vgl. WEYNS et al. (2007), S. 15.

⁷¹⁵ Vgl. SUDEIKAT (2010), S. 18.

⁷¹⁶ Vgl. SUDEIKAT (2010), S. 18.

⁷¹⁷ Vgl. SUDEIKAT (2010), S. 19.

⁷¹⁸ Vgl. SUDEIKAT (2010), S. 19.

⁷¹⁹ Vgl. SUDEIKAT (2010), S. 19.

⁷²⁰ Vgl. SUDEIKAT (2010), S. 19.

⁷²¹ Vgl. SUDEIKAT (2010), S. 19.

⁷²² Vgl. SUDEIKAT (2010), S. 19.

⁷²³ Vgl. SUDEIKAT (2010), S. 19.

4. **statisch oder dynamisch:** Dynamische Umgebungen sind Zustandsänderungen unterworfen, welche unabhängig von den Erwägungen der Agenten auftreten.⁷²⁴ Dies bedeutet, dass sich die Umgebung ändern kann, während der Agent seine nächste Aktion erwägt.⁷²⁵
5. **diskret oder kontinuierlich:** In diskreten Umgebungen sind lediglich eine begrenzte, klar definierte Anzahl an Wahrnehmungen und Aktionen möglich, wie bspw. bei einem Brettspiel.⁷²⁶ In kontinuierlichen Umgebungen erfahren dieselben Wahrnehmungen und Aktionen kontinuierliche Abstufungen.⁷²⁷

Die Umgebung wird als selbstständiges Element im Rahmen des Entwurfs von Multi-Agenten-Systemen verstanden.⁷²⁸ Eine Umgebung stellt einen gemeinsamen Raum für Agenten und Nicht-Agenten dar und strukturiert das Multi-Agenten-System physikalisch, kommunikativ und sozial.⁷²⁹ *Physikalische Strukturen* spiegeln die physikalische Verteilung der Agenten wieder und stellen ihnen Informationen zur räumlichen Verteilung zur Verfügung.⁷³⁰ *Kommunikative Strukturen* beschränken die Datenübertragung zwischen Agenten.⁷³¹ Übertragungen basieren auf einer direkten Übertragung unter Agenten oder einer indirekten Verbreitung, bei der die Umgebung Prozesse zur Vermittlung der Informationen bereitstellt.⁷³² *Soziale Strukturen* unterteilen die Agenten nach organisationalen Aspekten, wie bspw. Rollen oder Gruppen.⁷³³

Abbildung 10 stellt die Umgebung eines Multi-Agenten-Systems schematisch dar.

⁷²⁴ Vgl. SUDEIKAT (2010), S. 19.

⁷²⁵ Vgl. SUDEIKAT (2010), S. 19.

⁷²⁶ Vgl. SUDEIKAT (2010), S. 19.

⁷²⁷ Vgl. SUDEIKAT (2010), S. 19.

⁷²⁸ Vgl. SUDEIKAT (2010), S. 19.

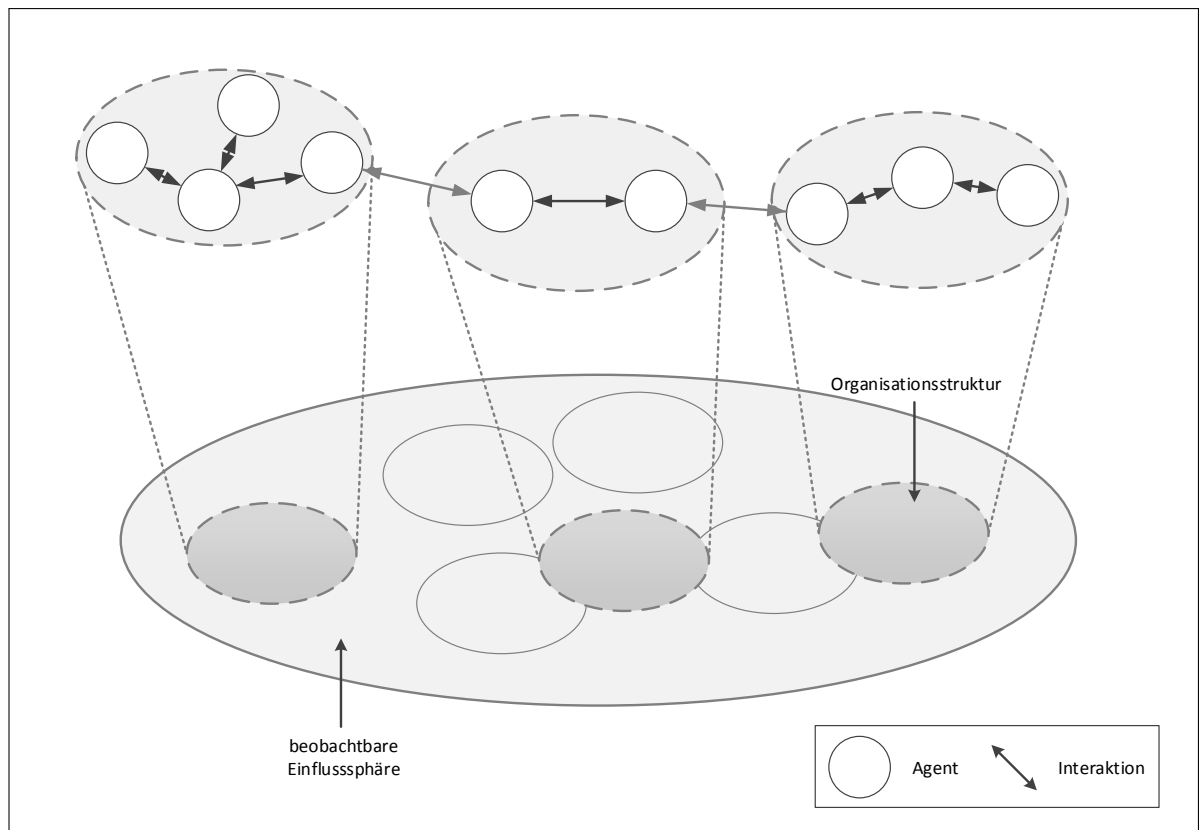
⁷²⁹ Vgl. SUDEIKAT (2010), S. 19.

⁷³⁰ Vgl. SUDEIKAT (2010), S. 19.

⁷³¹ Vgl. SUDEIKAT (2010), S. 19.

⁷³² Vgl. SUDEIKAT (2010), S. 19 f.

⁷³³ Vgl. SUDEIKAT (2010), S. 20.

Abbildung 10: Umgebung in Multi-Agenten-Systemen⁷³⁴

Umgebungen können *Ressourcen* und *Funktionalitäten* in Form von Nicht-Agenten bereitstellen.⁷³⁵ Ressourcen sind in diesem Zusammenhang Elemente, welche einen Zustand besitzen, der von den Agenten wahrgenommen und modifiziert werden kann.⁷³⁶ Funktionalitäten sind Entitäten, welche Agenten Aktionen bereitstellen, bspw. das Schreiben in eine Datenbank oder netzwerkbasierte Kommunikation.⁷³⁷

Die *lokale Beobachtbarkeit der Umgebung* ergibt sich daraus, wie die Agenten ihre Umwelt wahrnehmen.⁷³⁸ Der physikalische, kommunikative und soziale Kontext des Agenten schränkt die Beobachtungen ein.⁷³⁹

Traditionell gilt die implizite Hypothese, dass sich alle Agenten im Rahmen eines Multi-Agenten-Systems eine Umgebung teilen.⁷⁴⁰ Diese Annahme gilt auch im Rahmen dieser Forschungsarbeit.

Es muss jedoch festgestellt werden, dass GOUAÏCH und MICHEL aufzeigen, unter welchen Umständen auch Fälle von Multi-Agenten-Systemen mit mehreren Umgebungen denk-

⁷³⁴ Quelle: eigene Darstellung in Anlehnung an JENNINGS (2001), S. 37.

⁷³⁵ Vgl. SUDEIKAT (2010), S. 20.

⁷³⁶ Vgl. SUDEIKAT (2010), S. 20.

⁷³⁷ Vgl. SUDEIKAT (2010), S. 20.

⁷³⁸ Vgl. SUDEIKAT (2010), S. 20.

⁷³⁹ Vgl. SUDEIKAT (2010), S. 20.

⁷⁴⁰ Vgl. SUDEIKAT (2010), S. 20.

bar sind.⁷⁴¹ Auf diese Weise können für komplexe Multi-Agenten-Systeme verschiedene Aspekte der Applikationsdomäne, bspw. die Inter-Agenten-Kommunikation, in separaten Umgebungen entworfen werden.⁷⁴² Die Beziehung zwischen einem Agenten und einer Umgebung wird in diesem Fall charakterisiert durch die *Ontologie der Umgebung*, *Wahrnehmungsmöglichkeiten*, *aktive Möglichkeiten*, *Interaktionsfunktionen* sowie *Lokalisierungsfunktionen*.⁷⁴³

5.2.6 Konzeptionelle Modelle

In der agentenorientierten Programmierung existiert eine Vielzahl an Modellen, Methoden, Entwicklungskonzepten und Werkzeugen.⁷⁴⁴ Konzeptionelle Modelle werden primär für die Dokumentation von Methoden sowie die Bereitstellung von integrativen Agentensprachen genutzt und beinhalten Entwicklungskonzepte mit einer detaillierten Semantik.⁷⁴⁵ Die Zusammenführung von verschiedenen konzeptionellen Modellen in einem einheitlichen Ansatz stellt für sich genommen bereits eine wissenschaftliche Herausforderung dar, wie BEYDOUN et al. zeigen.⁷⁴⁶

Abbildung 11 fasst die Struktur des beschriebenen Multi-Agenten-Systems zusammen.

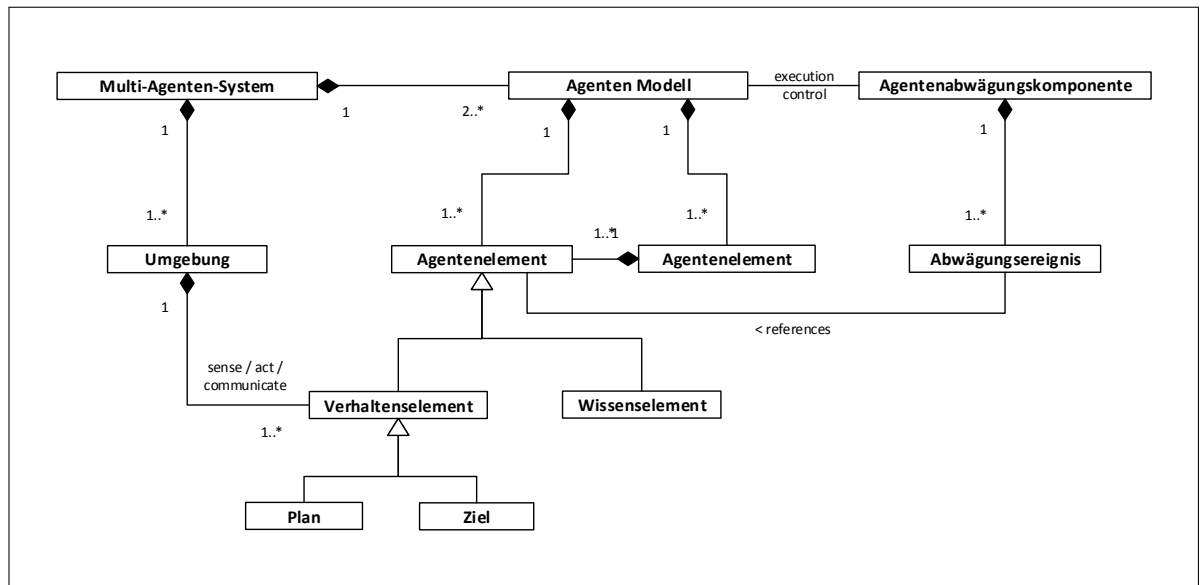


Abbildung 11: Konzeptionelle Struktur eines Multi-Agenten-Systems⁷⁴⁷

⁷⁴¹ Vgl. GOUAÏCH/MICHEL (2005), S. 423.

⁷⁴² Vgl. GOUAÏCH/MICHEL (2005), S. 423.

⁷⁴³ Vgl. GOUAÏCH/MICHEL (2005), S. 423.

⁷⁴⁴ Vgl. SUDEIKAT (2010), S. 22.

Einen Überblick über die existierenden Modelle und Methoden geben BEYDOUN et al. sowie BORDINI et al. Vgl. BEYDOUN et al. (2009), sowie BORDINI et al. (2006).

In der Fachliteratur finden sich zahlreiche Beispiele für Entwicklungskonzepte und Werkzeuge. Vgl. FORTINO/RUSSO (2012), COSSENTINO et al. (2010), BRESCIANI et al. (2002), JENNINGS (2001), sowie BURMEISTER et al. (1997).

⁷⁴⁵ Vgl. SUDEIKAT (2010), S. 22.

⁷⁴⁶ Vgl. BEYDOUN et al. (2009), S. 1 ff.

Multi-Agenten-Systeme enthalten Umgebungen und Agenten, welche ihrerseits Agentenelemente enthalten.⁷⁴⁸ Die Ausgestaltung der Agentenelemente hängt von der Agentenarchitektur ab.⁷⁴⁹ Dennoch lassen sich zwei Arten von Agentenelementen unterscheiden: *Wissenselemente*, welche lokal für den Agenten verfügbare Informationen repräsentieren, sowie *Verhaltenselemente*, welche die Vorgehensweisen und Aktivitäten repräsentieren, die ein Agent anwenden kann, wie bspw. Ziele und Pläne.⁷⁵⁰ Neben internen Funktionen innerhalb des Agenten übernehmen diese Elemente auch externe Funktionen, wie bspw. mit der direkten Umgebung zu interagieren oder mit anderen Agenten zu kommunizieren.⁷⁵¹ Agenten können zudem intern in Module aufgeteilt sein, die wiederum Agentenelemente enthalten.⁷⁵²

Die Steuerung eines Agenten, wie bspw. das Modifizieren von Wissen oder das Auslösen von Verhalten, wird von einem Abwägungsmechanismus kontrolliert, der sich in der *Agentenabwägungskomponente*⁷⁵³ befindet.⁷⁵⁴ Die Abwägung wird von der Ausführungsplattform basierend auf der Agentenarchitektur bereitgestellt und durch *Abwägungsereignisse* ausgelöst.⁷⁵⁵ Diese Ereignisse kennzeichnen Vorkommnisse, welche von den Agenten verarbeitet werden müssen.⁷⁵⁶ Sie beschreiben externe Einflüsse, wie bspw. das Empfangen von Nachrichten von anderen Agenten, sowie interne Zustandsänderungen, wie bspw. das Aktivieren von Zielen.⁷⁵⁷ Die Ereignisse referenzieren die Agentenelemente, wie bspw. Pläne, die durch das Ereignis betroffen sind.⁷⁵⁸

Das hier beschriebene, minimalistische Modell beschreibt lediglich eine generische Struktur, welche Details bezüglich der Agentenarchitektur und anderer Entwurfsaspekte vernachlässigt.⁷⁵⁹ Die Agentenelemente können um diese Details erweitert werden, wie das Beispiel der BDI-Architektur zeigt.⁷⁶⁰ BDI-Agenten enthalten zwei Arten von Verhaltenselementen: Ziele und Pläne. Ziele lassen sich, falls erforderlich, den individuellen Anforderungen entsprechend weiter unterteilen in verschiedene Zielarten.⁷⁶¹

⁷⁴⁷ Quelle: eigene Darstellung in Anlehnung an SUDEIKAT (2010), S. 23.

⁷⁴⁸ Vgl. SUDEIKAT (2010), S. 22.

⁷⁴⁹ Vgl. SUDEIKAT (2010), S. 22.

⁷⁵⁰ Vgl. SUDEIKAT (2010), S. 22.

⁷⁵¹ Vgl. SUDEIKAT (2010), S. 22.

⁷⁵² Vgl. SUDEIKAT (2010), S. 22.

⁷⁵³ Im Englischen „agent reasoning component“. Vgl. SUDEIKAT (2010), S. 22.

⁷⁵⁴ Vgl. SUDEIKAT (2010), S. 22.

⁷⁵⁵ Vgl. SUDEIKAT (2010), S. 22.

⁷⁵⁶ Vgl. SUDEIKAT (2010), S. 22 f.

⁷⁵⁷ Vgl. SUDEIKAT (2010), S. 22 f.

⁷⁵⁸ Vgl. SUDEIKAT (2010), S. 23.

⁷⁵⁹ Vgl. SUDEIKAT (2010), S. 23.

⁷⁶⁰ Vgl. SUDEIKAT (2010), S. 23.

⁷⁶¹ Vgl. SUDEIKAT (2010), S. 23.

Die Klassifizierung von Verhaltensweisen ist für die Entwicklung individueller Agenten entscheidend.⁷⁶² Einige Entwicklungswerkzeuge unterstützen das Festlegen von Verhaltensweisen, jedoch bleibt die Identifizierung und Implementierung der Verhaltensweisen eine konzeptionelle Anforderung, die von den bei der Entwicklung tatsächlich verfügbaren Konzepten und Anforderungen abhängt.⁷⁶³

Im Folgenden ist die Klassifizierung von Verhaltensweisen anhand der bereits angesprochenen BDI-Architektur beispielhaft erläutert.⁷⁶⁴ Die Aktivitäten von BDI-Agenten folgen den Konzepten der Zielerwägung und Abwägung, welche zur Aktivierung von Plänen und Zielen individueller Agenten führen.⁷⁶⁵ Regelmäßig stellen Pläne verschiedene Wege zur Erreichung desselben Ziels dar, so dass der Agent zwischen verschiedenen Plänen zur Erreichung seines Ziels wählen muss.⁷⁶⁶ Auch die wiederholte oder sequentielle Ausführung von Plänen kann zur Erreichung eines Zieles notwendig sein.⁷⁶⁷ Der Abwägungsmechanismus des Agenten wählt die zu erreichenden Ziele aus und aktiviert anschließend die hierfür ausgewählten Pläne.⁷⁶⁸ Die Ausführung der Pläne kann weitere „Unterziele“ vorsehen, deren Aktivierungen ihrerseits die Aktivierung weiterer „Unterpläne“ zur Folge haben.⁷⁶⁹ Daher kann der Entwurf des Agentenverhaltens als Ziel-Plan-Baum⁷⁷⁰ verstanden werden.⁷⁷¹ Die Abzweigungen eines Baums stellen die verschiedenen zur Auswahl stehenden Ziele dar, welche sich in Pläne verzweigen, die zu ihrer Ausführung geeignet sind.⁷⁷² Die Pläne können sich ihrerseits wieder in Unterziele verzweigen, die wiederum Unterpläne zu ihrer Erreichung erfordern.⁷⁷³

Abbildung 12 illustriert diesen Sachverhalt und verdeutlicht, dass entweder alternative Optionen zur Erreichung eines Ziels oder Plans zur Verfügung stehen (XOR-Verknüpfung) oder aber die erfolgreiche Ausführung aller Elemente eines Plans oder Ziels zur deren Erfüllung notwendig ist (UND-Verknüpfung).⁷⁷⁴

⁷⁶² Vgl. SUDEIKAT (2010), S. 23.

⁷⁶³ Vgl. SUDEIKAT (2010), S. 23.

⁷⁶⁴ Vgl. SUDEIKAT (2010), S. 23.

⁷⁶⁵ Vgl. SUDEIKAT (2010), S. 23.

⁷⁶⁶ Vgl. SUDEIKAT (2010), S. 23.

⁷⁶⁷ Vgl. SUDEIKAT (2010), S. 23.

⁷⁶⁸ Vgl. SUDEIKAT (2010), S. 23.

⁷⁶⁹ Vgl. SUDEIKAT (2010), S. 23.

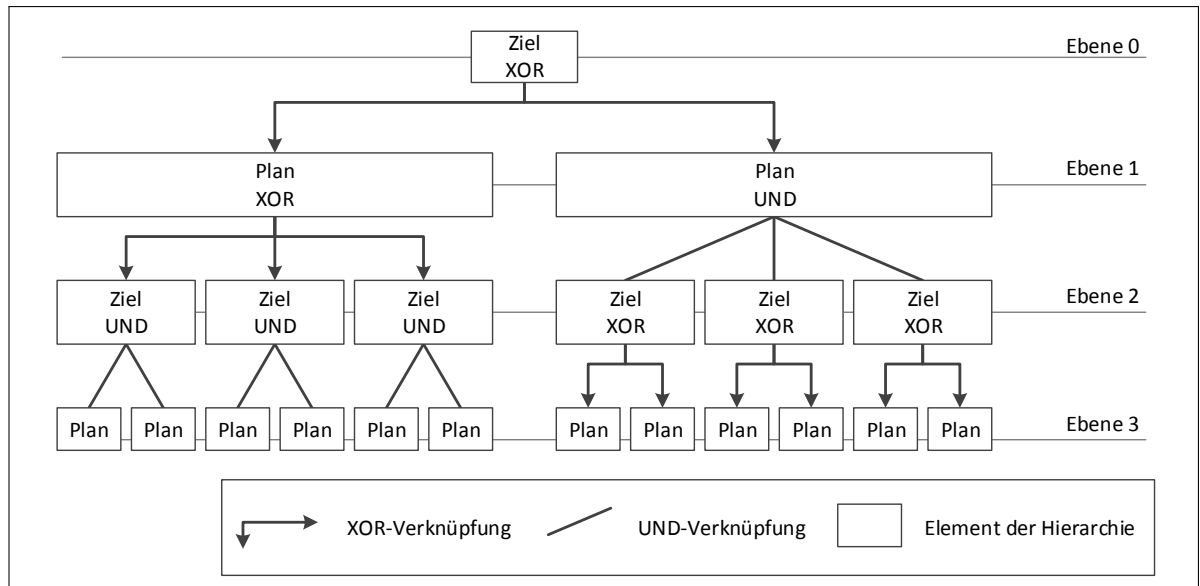
⁷⁷⁰ Im Englischen „goal-plan tree“. Vgl. SUDEIKAT (2010), S. 23.

⁷⁷¹ Vgl. SUDEIKAT (2010), S. 23.

⁷⁷² Vgl. SUDEIKAT (2010), S. 23.

⁷⁷³ Vgl. SUDEIKAT (2010), S. 23.

⁷⁷⁴ Vgl. SUDEIKAT (2010), S. 23.

Abbildung 12: Generisches Beispiel eines Ziel-Plan-Baums⁷⁷⁵

Ziele stellen in diesem Zusammenhang kein absolutes Konzept dar, sondern dienen vielmehr der Abstraktion des Kontexts, in welchem die Ausführung von Plänen zulässig ist.⁷⁷⁶ Die Pläne werden nicht zur Laufzeit generiert, sondern während der Entwicklung der Agenten festgelegt.⁷⁷⁷ Daher kann die prozedurale Logik sowohl die Auswahl alternativer Unterziele als auch die sequentielle Aktivierung weiterer Unterzielen beschreiben.

Während UND-Verknüpfungen wie beschrieben die sequentielle Ausführung der zugehörigen Elemente fordern, stellen XOR-Verknüpfungen den Agenten vor die Aufgabe, zwischen Alternativen abzuwägen.⁷⁷⁸ Hierdurch lassen sich klar getrennte Verhaltensweisen identifizieren, welche eine Hierarchie von Verhaltensweisen bilden.⁷⁷⁹ Die „Ebene 0“ in Abbildung 12 stellt hierbei die höchste Ebene an Zielen dar.⁷⁸⁰ Die „Ebene 1“ bildet die zur Erreichung dieser Ziele zur Verfügung stehenden Basis-Aktivitäten ab.⁷⁸¹ Die Verarbeitung dieser Hierarchie und der Grad ihrer Verzweigung und Granularität entscheidet über die Struktur der internen Agentenelemente.⁷⁸²

⁷⁷⁵ Quelle: eigene Darstellung in Anlehnung an SUDEIKAT (2010), S. 24.

⁷⁷⁶ Vgl. SUDEIKAT (2010), S. 24.

⁷⁷⁷ Vgl. SUDEIKAT (2010), S. 24.

⁷⁷⁸ Vgl. SUDEIKAT (2010), S. 24.

⁷⁷⁹ Vgl. SUDEIKAT (2010), S. 24.

⁷⁸⁰ Vgl. SUDEIKAT (2010), S. 24.

⁷⁸¹ Vgl. SUDEIKAT (2010), S. 24.

⁷⁸² Vgl. SUDEIKAT (2010), S. 24.

5.2.7 Koordination in Multi-Agenten-Systemen

Die Koordination von Agenten ist ein wichtiger Aspekt der Entwicklung von Multi-Agenten-Systemen.⁷⁸³ Die Kooperation und Kollaboration von Agenten muss durch die Entwickler geplant werden, damit ein funktionierendes System autonomer Software-Agenten entsteht.⁷⁸⁴ So kann das Multi-Agenten-System dem Konzept eines selbstorganisierenden, dynamischen Systems autonomer Akteure gerecht werden.⁷⁸⁵ Zur Erreichung dieses Ziels stehen viele Koordinationsmechanismen bereit, wie bspw. Organisationsstrukturen, Verhandlungen, Auktionen, Kontrakte und verteiltes Planen.⁷⁸⁶

Wegen der Vielfältigkeit der Koordinationsmechanismen ist die Koordination in Multi-Agenten-Systemen ein interdisziplinäres und unscharf definiertes Konzept, zu dessen Umsetzung eine Vielzahl von Techniken zur Verfügung stehen.⁷⁸⁷ MALONE und CROWSTON definieren Koordination als das „Leiten von Abhängigkeiten zwischen Aktivitäten“. ⁷⁸⁸ Diese Sichtweise abstrahiert von den angewandten Mechanismen und betont die Notwendigkeit die Handlungsmöglichkeiten eines einzelnen Agenten zu beschränken, damit das Multi-Agenten-System koordinierte Aktivitäten zeigen kann.⁷⁸⁹

Die konzeptionelle Unterscheidung und technische Trennung von Berechnungen⁷⁹⁰ und Koordination ist ebenfalls von Bedeutung:⁷⁹¹ Berechnungen bezeichnen hier die Aktivitäten, welche durch die Software schrittweise ausgeführt werden.⁷⁹² Koordination meint im Gegensatz dazu die Kollaboration von Berechnungen. Die getroffene Unterscheidung ermöglicht die Wiederverwendung der berechnenden Softwarekomponenten und koordinierenden Mechanismen.⁷⁹³

Ein Koordinationsmodell besteht aus drei Komponenten: einer Menge von Entitäten, welche koordiniert werden sollen, einer Menge von Medien, welche zur Koordinierung eingesetzt werden, und einer Beschreibung des Modells.⁷⁹⁴ *Koordinations-sprachen* erlauben die Beschreibung auszuführender Koordinationsmaßnahmen, bspw. durch eine Kontrolle der zur Koordination nötigen Aktivitäten und Kommunikation.⁷⁹⁵ Es kann argumentiert

⁷⁸³ Vgl. SUDEIKAT (2010), S. 24.

⁷⁸⁴ Vgl. SUDEIKAT (2010), S. 24.

⁷⁸⁵ Vgl. SUDEIKAT (2010), S. 24.

⁷⁸⁶ Vgl. BEDROUNI et al. (2009), S. 75 ff.

⁷⁸⁷ Vgl. BEDROUNI et al. (2009), S. 2 ff. u. 125 ff.

⁷⁸⁸ Vgl. MALONE/CROWSTON (1994), S. 90.

⁷⁸⁹ Vgl. SUDEIKAT (2010), S. 24.

⁷⁹⁰ Im Englischen „computation“. Vgl. SUDEIKAT (2010), S. 25.

⁷⁹¹ Vgl. SUDEIKAT (2010), S. 25.

⁷⁹² Vgl. SUDEIKAT (2010), S. 25.

⁷⁹³ Vgl. BEDROUNI et al. (2009), S. 4 f.

⁷⁹⁴ Vgl. PAPADOPOULOS/ARBAB (1998), S. 4.

⁷⁹⁵ Vgl. PAPADOPOULOS/ARBAB (1998), S. 4.

werden, dass Koordinationsstrukturen grundlegende Abstraktionen darstellen, welche den Agenten Funktionalität in Form von Koordination bereitstellen.⁷⁹⁶ Diese Argumentation korrespondiert mit der Feststellung, dass Umgebungen als Abstraktionen und Bereitsteller von Ressourcen und Funktionalitäten gesehen werden können.⁷⁹⁷

Der Begriff „Medien“ schließt die Umgebung und Infrastruktur eines Multi-Agenten-Systems mit ein.⁷⁹⁸ Medien können in datengetriebene und kontrollgetriebene Medien unterschieden werden.⁷⁹⁹

Datengetriebene Medien entkoppeln Systemkomponenten und stellen eine Infrastruktur zum Austausch bereit, welche die Systemkomponenten in die Lage versetzen, autonom koordinationsbezogene Aktivitäten ausführen.⁸⁰⁰ Die Systemkomponenten können die ausgetauschten Daten lokal verarbeiten und weitere Koordinationsmechanismen aktivieren.⁸⁰¹ Dies zeigt, dass eine Trennung von berechnenden und koordinierenden Aspekten nicht stattfindet.⁸⁰² Die Vermeidung der Vermischung dieser Aspekte fällt in den Aufgabenbereich der Softwareentwicklung.⁸⁰³

Kontrollgetriebene Medien erzwingen diese Trennung durch neuartige Programmiersprachen, welche die koordinierten Entitäten mit Hilfe von Schnittstellen verbinden.⁸⁰⁴ Die Entitäten werden als „black boxes“ konzipiert, deren Koordination in externen Sprachmodellen festgelegt wird.⁸⁰⁵ Ein regelmäßig angewendetes Koordinationsmodell ist das Modell ereignisbasierter Verarbeitung, bei der Zustandsänderungen beobachtet und an die Entitäten kommuniziert werden.⁸⁰⁶

Die Selbstorganisation von Multi-Agenten-Systemen kann aus einer pragmatischen Perspektive als Form dezentraler Koordination verstanden werden.⁸⁰⁷

⁷⁹⁶ Vgl. OMICINI et al. (2004), 286 f.

⁷⁹⁷ Vgl. Kapitel 5.2.5, S. 85.

⁷⁹⁸ Vgl. SUDEIKAT (2010), S. 25.

⁷⁹⁹ Vgl. PAPADOPOULOS/ARBAB (1998), S. 4 ff.

⁸⁰⁰ Vgl. PAPADOPOULOS/ARBAB (1998), S. 3 ff.

⁸⁰¹ Vgl. PAPADOPOULOS/ARBAB (1998), S. 5.

⁸⁰² Vgl. SUDEIKAT (2010), S. 25.

⁸⁰³ Vgl. SUDEIKAT (2010), S. 25.

⁸⁰⁴ Vgl. PAPADOPOULOS/ARBAB (1998), S. 4 ff.

⁸⁰⁵ Vgl. PAPADOPOULOS/ARBAB (1998), S. 3 f.

⁸⁰⁶ Vgl. PAPADOPOULOS/ARBAB (1998), S. 36 f.

⁸⁰⁷ Vgl. SUDEIKAT (2010), S. 25.

5.2.8 Implementierung agentenbasierter Softwaresysteme

Das Konzept agentenbasierter Softwareentwicklung unterscheidet sich von traditionellen Softwareentwicklungskonzepten durch den Einsatz von Agenten und Umgebungen.⁸⁰⁸ SUDEIKAT stellt fest, dass zur Konzeption der Softwarekomponenten spezielle Werkzeuge bereitgestellt werden müssen, während zur Realisierung der Software bewährte Techniken wie die objektorientierte Programmierung eingesetzt werden können.⁸⁰⁹

Die Adaption von Organisationsstrukturen bei der Entwicklung von Multi-Agenten-Systemen ist ein Beispiel.⁸¹⁰ Diese Konzepte beschreiben das Verhalten von Agenten sowie ihren sozialen Kontext, wie bspw. im Fall von Gruppen und Rollen im Rahmen des „Agent/Group/Role“-Modells.⁸¹¹ Das Rollenkonzept wird benutzt, um Agentenverhalten zu charakterisieren, wie bspw. die Beschreibung von interagierenden Akteuren mit Hilfe von Interaktionsprotokollen.⁸¹² Gruppen werden als Partitionen einer Organisationsstruktur verstanden. Dieser Ansatz ermöglicht die Beschreibung eines Multi-Agenten-Systems anhand seiner Struktur und der Erwartungen an das Verhalten seiner Agenten.⁸¹³

Kritisch zu sehen ist die Einordnung von agentenorientierten Ansätzen als neue Art der Softwareentwicklung, welche als zuverlässiger und agiler dargestellt wird als bspw. objektorientierte Ansätze.⁸¹⁴ Es darf nicht unterschlagen werden, dass die agentenorientierte Programmierung eine Weiterentwicklung des Paradigmas der objektorientierten Programmierung ist. Die als Agenten konzipierten Softwarekomponenten können jedoch trotz der ihnen zugeschriebenen Eigenschaften als Objekte im Sinne der objektorientierten Programmierung verstanden werden. Eine vielfach publizierte Fehlkonzeption der objektorientierten Programmierung ist, dass ihr wesentlicher Beitrag zur Softwareentwicklung in den Konzepten von Objekten und Klassen besteht.

ALAN KAY, der Erfinder des Begriffs „objektorientiert“ und der objektorientierten Programmiersprache *Smalltalk*, definierte Objektorientierung im Rahmen von Smalltalk folgendermaßen:⁸¹⁵

⁸⁰⁸ Vgl. SUDEIKAT (2010), S. 26.

⁸⁰⁹ Vgl. SUDEIKAT (2010), S. 27.

⁸¹⁰ Vgl. FERBER et al. (2004), S. 214.

⁸¹¹ Vgl. FERBER et al. (2004), S. 215 f.

⁸¹² Vgl. FERBER et al. (2004), S. 215 f.

⁸¹³ Vgl. SUDEIKAT (2010), S. 27.

⁸¹⁴ Vgl. SUDEIKAT (2010), S. 36 ff.

⁸¹⁵ Vgl. KAY (1993), S. 27 f.

1. Alles ist ein Objekt.⁸¹⁶
2. Objekte kommunizieren durch das Senden und Empfangen von Nachrichten, welche ihrerseits Objekte darstellen.
3. Objekte haben ihren eigenen, als Objekte strukturierten Speicher.
4. Jedes Objekt ist die Instanz einer Klasse, welche selbst ein Objekt darstellt.
5. Die Klasse beinhaltet das Verhalten aller ihrer Instanzen (in der Form von Objekten in einer Programmliste).
6. Um eine Programmliste auszuführen, wird die Ausführungskontrolle dem ersten Objekt gegeben und das verbleibende Programm als dessen Nachricht behandelt.

KAY bereute später die Wahl des Begriffs „Objektorientierung“ und schrieb, dass die objektorientierte Programmierung und Smalltalk nicht durch die Syntax oder das Klassenkonzept gekennzeichnet seien.⁸¹⁷ Er bereue, dass er den Begriff „Objekt“ geprägt hatte⁸¹⁸, da es viele Menschen dazu brachte, sich auf das weniger relevante Konzept seiner Idee (die Objekte) zu konzentrieren.⁸¹⁹ Er selbst betrachtete den Aspekt des Austauschs von Nachrichten zwischen den Objekten als den wesentlich wichtigeren Teil seiner Idee.⁸²⁰

Diese Ausführungen verdeutlichen, dass die agentenorientierte Programmierung in der Art, wie sie in vielen Publikationen dargestellt wird, streng genommen keine Weiterentwicklung des Paradigmas der objektorientierten Programmierung ist, sondern die konsequente Umsetzung dessen ursprünglicher Idee darstellt.

5.2.9 Relevanz für die wissenschaftlichen Probleme

Der Einsatz von Multi-Agenten-Systemen eignet sich besonders in geographisch und funktional weit verteilten Problemdomänen, deren Subsysteme einen hohen Grad an Autonomie aufweisen und in einem dynamischen Umfeld angesiedelt sind.⁸²¹ Die Domäne „Verkehr und Transport“ sowie der Güterverkehr entsprechen diesen Kriterien.⁸²²

⁸¹⁶ Diese Formulierung von KAY ist kritisch zu hinterfragen, da bspw. Aktionen, wie das Senden und Empfangen von Nachrichten, keine Objekte sind. KAY selbst bedauerte seine Wortwahl später. Vgl. KAY (1998), o.S.

⁸¹⁷ Vgl. KAY (1998), o.S.

⁸¹⁸ Vgl. KAY (1993), S. 1.

⁸¹⁹ Vgl. KAY (1998), o.S.

⁸²⁰ Vgl. KAY (1998), o.S.

⁸²¹ Vgl. BURMEISTER et al. (1997), S. 3.

⁸²² Vgl. BURMEISTER et al. (1997), S. 3 f., sowie Anhang, S. 289.

Die zur Schaffung einer dezentral organisierten Online-Frachtenbörse notwendigen Software-Agenten müssen in der Lage sein, ihre Umgebung zu beobachten und miteinander zu kommunizieren. Die Umgebung muss Informationen über Fahrpläne, Streckenverfügbarkeiten u.a. zur Verfügung stellen. Die implementierten Agenten sind hybride Agenten, da sie sowohl reaktiv auf am Markt bestehende Angebote reagieren als auch in der Lage sein müssen, selbst Angebote in den Markt einzustellen. Hierzu müssen den Agenten bereits zur Entwicklungszeit entsprechende Pläne implementiert werden. Der Koordination der Agenten untereinander kommt wegen des dezentralen Aspekts ebenso eine erhöhte Bedeutung zu wie den hierzu verwendeten Beschreibungs- und Koordinationssprachen sowie den eingesetzten Medien.

Es wurde festgestellt, dass die agentenorientierte Gestaltung von Softwaresystemen seit Jahrzehnten ein populäres Thema in der Fachliteratur ist. Die Voraussage, dass sich neue, agentenorientierte Programmiersprachen bilden werden, konnte jedoch – anders als bei den zum Vergleich angeführten funktionalen und objektorientierten Ansätzen in den Jahrzehnten zuvor – nicht beobachtet werden. Vielmehr findet die Gestaltung von agentenorientierten Softwaresystemen auf einer höheren Abstraktionsebene statt und wird mit Hilfe von bestehenden, zumeist objektorientierten Programmiersprachen realisiert.

5.3 Auktionen

5.3.1 Einführung

MCAFEE und MCMILLAN definieren eine Auktion als eine marktliche Institution, welche die Preise und Allokation von Gütern auf Basis von Geboten der Marktteilnehmer nach im Vorhinein festgelegten Regeln determiniert.⁸²³

Auktionen werden seit Jahrtausenden von Menschen benutzt, um Preise für Güter zu bestimmen.⁸²⁴ Sie werden stets dann verwendet, wenn bei den Verkäufern eines Gutes Ungewissheit über die Zahlungsbereitschaft ihrer Käufer besteht.⁸²⁵ Hätten die Verkäufer Gewissheit über den erzielbaren Preis, so würden sie ihr Gut regulär zu diesem oder einem geringfügig darunterliegenden Preis anbieten.⁸²⁶ Die Tatsache, dass sowohl Verkäufer als auch Käufer sich bezüglich des erlösbaren Verkaufspreises nicht sicher sind, kann somit als inhärentes Charakteristikum von Auktionen bezeichnet werden.⁸²⁷

⁸²³ Vgl. MCAFEE/MCMILLAN (1987), S. 701.

⁸²⁴ Vgl. MILGROM/WEBER (1982), S. 1089.

⁸²⁵ Vgl. KRISHNA (2009), S. 2.

⁸²⁶ Vgl. KRISHNA (2009), S. 2.

⁸²⁷ Vgl. KRISHNA (2009), S. 2.

In Anknüpfung an Kapitel 5.1 stellen Online-Auktionsplattformen eine Sonderform von elektronischen Marktplätzen mit dynamischer Preisfestsetzung dar, auf denen verschiedenste Arten von Auktionen – steigende, fallende, offene, verdeckte, nachfrager- und anbieterseitige Gebote – durchgeführt werden.⁸²⁸ Auktionen dienen der Versteigerung von sowohl materiellen als auch immateriellen Gütern: Nutztiere, Bürobedarf, Antiquitäten und Kunstwerke werden ebenso versteigert wie UMTS-Frequenzen und Verlagsrechte.⁸²⁹

Die jeweiligen privaten Bewertungen der Käufer werden durch den Wettbewerb teilweise offengelegt.⁸³⁰ Wann und wie durch eine Auktion eine Transaktion zustande kommt, legt das Auktionsmodell fest.⁸³¹

Zur Unterscheidung und Abgrenzung verschiedener Auktionsmodelle lassen sich einige Kriterien identifizieren:

1. Ein erstes Kriterium, um Auktionsmodelle zu unterscheiden, ist die Anzahl der versteigerten Güter.⁸³² Eingutauktionen bilden den einfachsten Fall: Ein – in der Regel unteilbares – Gut wird von einem Verkäufer unter mehreren potentiellen Käufern versteigert.⁸³³ Mehrgutauktionen lassen im Vergleich dazu komplexere Konfigurationen zu.⁸³⁴ So können die versteigerten Güter in ihrer Beschaffenheit identisch sein, wie bspw. bei der Auktion von Bürobedarf. Sind die versteigerten Güter nicht identisch, so ist es möglich, dass es sich bei ihnen um Substitute oder Komplemente handelt.⁸³⁵ Eine weitere Möglichkeit ist, dass der oder die Verkäufer ihre Güter nur in Bündeln – oder auch synonym: Kombinationen – veräußern wollen.⁸³⁶
2. Ein zweites Kriterium, um Auktionsmodelle zu unterscheiden, ist die Art der Gebotsabgabe.⁸³⁷ Sie kann offen oder verdeckt erfolgen.⁸³⁸ Bei der offenen Gebotsabgabe erhalten alle Teilnehmer der Auktion Kenntnis der Gebote der anderen Teilnehmer. Bei der verdeckten Gebotsabgabe erhält lediglich der Auktionator Kenntnis der einzelnen Gebote.

⁸²⁸ Vgl. REBSTOCK (2000), S. 12, sowie Kapitel 5.1, S. 60.

⁸²⁹ Vgl. MILGROM/WEBER (1982), S. 1089.

⁸³⁰ Vgl. BICHLER et al. (2005), S. 126.

⁸³¹ Vgl. BICHLER et al. (2005), S. 126.

⁸³² Vgl. PANKRATZ (2003), S. 2.

⁸³³ Vgl. PANKRATZ (2003), S. 2.

⁸³⁴ Vgl. PANKRATZ (2003), S. 2.

⁸³⁵ Vgl. PANKRATZ (2003), S. 3.

⁸³⁶ Vgl. PANKRATZ (2003), S. 2 f.

⁸³⁷ Vgl. PANKRATZ (2003), S. 3.

⁸³⁸ Vgl. PANKRATZ (2003), S. 3.

3. Ein drittes Kriterium ist die Anzahl der Verkäufer.⁸³⁹ Bei einseitigen Auktionen versteigert ein Verkäufer ein Gut oder mehrere Güter an mehrere Käufer.⁸⁴⁰ Bei zweiseitigen Auktionen versteigern mehrere Verkäufer ein Gut oder mehrere Güter an mehrere Käufer.⁸⁴¹ Ein Beispiel für eine zweiseitige Auktion sind Börsen.⁸⁴²

Zudem können Auktionsmodelle aufgrund der von den Teilnehmern in Bezug auf die versteigerten Güter vorgenommenen Bewertungen unterschieden werden.

Das *independent private values model* beschreibt ein Auktionsmodell, bei dem ein unteilbares Gut an einen von mehreren Bietern verkauft werden soll.⁸⁴³ Jeder Bieter ist risikoneutral und kennt den Wert des Gutes für ihn selbst, hat jedoch keine Vorstellung, wie die anderen Bieter den Wert des Gutes für sich einschätzen.⁸⁴⁴

Das *common values model* beschreibt ein Auktionsmodell, bei dem ebenfalls ein unteilbares Gut an einen von mehreren Bietern verkauft werden soll.⁸⁴⁵ Im Gegensatz zum *independent private values model* hat das versteigerte Gut jedoch einen allgemeinen Wert, der von den teilnehmenden Bietern individuell geschätzt wird.⁸⁴⁶ Als Beispiel dient die Versteigerung von Rechten zur Förderung von Erdöl innerhalb einer bestimmten Region.⁸⁴⁷ Hierbei wäre der Wert dieser Rechte durch das förderbare Öl der Region gegeben, wenn dieser Wert sicher bekannt wäre.⁸⁴⁸ Da jedoch die Menge, die Qualität, der Aufwand der Förderung und der später erzielbare Verkaufspreis des Öls im Regelfall während der Auktion unbekannt sind, basiert die Zahlungsbereitschaft der Teilnehmer der Auktion auf ihren individuellen Schätzungen der zuvor genannten Charakteristika.⁸⁴⁹ Die Bieter haben voneinander unabhängige, private Bewertungen des allgemeinen Werts.⁸⁵⁰

Unabhängig davon, ob klassische oder kombinatorische Auktionsformen betrachtet werden, können einige wünschenswerte, ökonomische Eigenschaften von Auktionsmodellen identifiziert werden.⁸⁵¹ Das Auktionsmodell beschreibt die Regeln, nach denen die Auktion abläuft. Die teilnehmenden Bieter sollen angehalten werden, ihre Präferenzen so weit

⁸³⁹ Vgl. PANKRATZ (2003), S. 3.

⁸⁴⁰ Vgl. PANKRATZ (2003), S. 3.

⁸⁴¹ Vgl. PANKRATZ (2003), S. 3.

⁸⁴² Vgl. PANKRATZ (2003), S. 3.

⁸⁴³ Vgl. MILGROM/WEBER (1982), S. 1090.

⁸⁴⁴ Vgl. MILGROM/WEBER (1982), S. 1090.

⁸⁴⁵ Vgl. MILGROM/WEBER (1982), S. 1093.

⁸⁴⁶ Vgl. MILGROM/WEBER (1982), S. 1093.

⁸⁴⁷ Vgl. MILGROM/WEBER (1982), S. 1093.

⁸⁴⁸ Vgl. MILGROM/WEBER (1982), S. 1093.

⁸⁴⁹ Vgl. MILGROM/WEBER (1982), S. 1093.

⁸⁵⁰ Vgl. MILGROM/WEBER (1982), S. 1093.

⁸⁵¹ Vgl. BICHLER et al. (2005), S. 128.

offen zu legen, dass eine optimale Allokation der versteigerten Güter bzgl. ihrer privaten Bewertungen berechnet werden kann.⁸⁵²

Hierzu werden die beiden Optimalitätskriterien *allokative Effizienz* und *Ertragsmaximierung* diskutiert. Allokative Effizienz meint, dass der Nutzen aller teilnehmenden Bieter in der Allokation maximiert wird.⁸⁵³ Ertragsmaximierung meint, dass der Nutzen eines speziellen Teilnehmers, bspw. des Auktionators, in der Allokation maximiert wird.⁸⁵⁴

Zwei Regeln, welche teilnehmende Bieter dazu bewegen sollen, ihre Präferenzen offen zu legen, sind die *Anreizkompatibilität* und die *Strategiebeständigkeit*.⁸⁵⁵ Anreizkompatibilität meint, dass ein Bietverhalten entsprechend den eigenen, tatsächlichen Präferenzen ein BAYES-NASH-Gleichgewicht darstellt.⁸⁵⁶ Strategiebeständigkeit meint ein Gleichgewicht in dominanten Strategien: Jeder Bieter verhält sich optimal, unabhängig von der Strategiewahl der anderen Bieter.⁸⁵⁷ Ist Strategiebeständigkeit gegeben, so reduziert sich die strategische Komplexität für die teilnehmenden Bieter auf ein Minimum.⁸⁵⁸

Weitere Anforderungen an Auktionsmodelle sind *individuelle Rationalität* und *budgetäre Ausgeglichenheit*.⁸⁵⁹ Individuelle Rationalität meint, dass das Auktionsmodell allen Teilnehmern einen positiven erwarteten Nutzen im Vergleich zur Nichtteilnahme an der Auktion bieten muss.⁸⁶⁰ Budgetäre Ausgeglichenheit meint, dass auch der Auktionator durch die Auktion keinen Verlust erleiden darf.⁸⁶¹

In der klassischen Auktionstheorie wurden zahlreiche Varianten von Modellen mit unabhängigen oder auch mit gleichen Bewertungen der Bieter theoretisch und experimentell untersucht.⁸⁶² Die Ergebnisse aus Versuchen zu klassischen Auktionen sind jedoch nur eingeschränkt auf kombinatorische Auktionen übertragbar.⁸⁶³

⁸⁵² Vgl. BICHLER et al. (2005), S. 128.

⁸⁵³ Vgl. BICHLER et al. (2005), S. 128.

⁸⁵⁴ Teilweise liegt bzgl. der Preise ein Minimierungsziel vor, bspw. bei Beschaffungsauktionen. Dennoch wird in diesen Fällen der Nutzen eines Teilnehmers maximiert. Vgl. BICHLER et al. (2005), S. 128.

⁸⁵⁵ Vgl. BICHLER et al. (2005), S. 129.

⁸⁵⁶ Ein BAYES-NASH-Gleichgewicht liegt vor, wenn kein Bieter einen rationalen Anlass hat, eigenständig (also ohne Berücksichtigung von Kartellabsprachen mit anderen Bietern) von seinem Gleichgewichts-bietverhalten abzuweichen. Vgl. BICHLER et al. (2005), S. 129.

⁸⁵⁷ Vgl. BICHLER et al. (2005), S. 129.

⁸⁵⁸ Vgl. BICHLER et al. (2005), S. 129.

⁸⁵⁹ Vgl. BICHLER et al. (2005), S. 129.

⁸⁶⁰ Vgl. BICHLER et al. (2005), S. 129.

⁸⁶¹ Vgl. BICHLER et al. (2005), S. 129.

⁸⁶² Vgl. BICHLER et al. (2005), S. 129.

⁸⁶³ Vgl. BICHLER et al. (2005), S. 129.

5.3.2 Klassische Auktionsformen

Als Standard-Auktionen werden im Folgenden die vier verbreitetsten Formen der Auktion eines einzelnen Gutes verstanden.⁸⁶⁴

MILGROM und WEBER identifizieren vier Arten von Standard-Auktionen:

1. die offene englische Auktion,
2. die offene holländische Auktion,
3. die verdeckte Erstpreisauktion sowie
4. die verdeckte Zweitpreisauktion.⁸⁶⁵

Abbildung 13 zeigt den allen Standard-Auktionen gemeinsamen schematischen Aufbau:

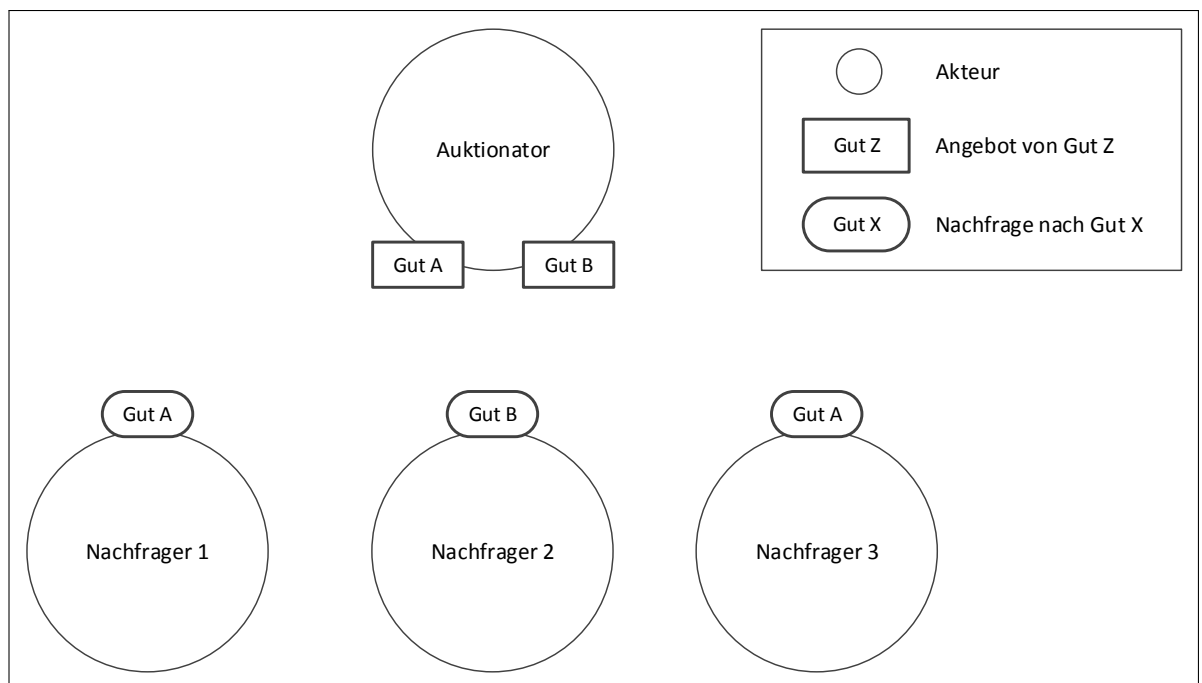


Abbildung 13: Grafisches Beispiel für eine Standard-Auktion⁸⁶⁶

Ein Auktionator versteigert eine zuvor festgelegte Menge an Gütern. Es wird stets nur ein Gut gleichzeitig versteigert. Alle an der Auktion teilnehmenden Nachfrager können Gebote auf jedes der Güter abgeben. Die verschiedenen Standard-Auktionen unterscheiden sich durch die Rahmenbedingungen für die Gebotsabgabe.

Die *englische Auktion* existiert in vielen Varianten.⁸⁶⁷ Eine mögliche Variante sieht folgendermaßen aus: Die Bieter auf ein Auktionsgut geben in jeder Bietrunde durch ein

⁸⁶⁴ MILGROM und WEBER sowie KRISHNA bezeichnen diese Standard-Auktionen als „common auction forms“. Vgl. KRISHNA (2009), S. 2, sowie MILGROM/WEBER (1982), S. 1089 f.

⁸⁶⁵ Vgl. MILGROM/WEBER (1982), S. 1089 f.

⁸⁶⁶ Quelle: eigene Darstellung.

⁸⁶⁷ Vgl. MILGROM/WEBER (1982), S. 1103 f.

zuvor verabredetes Zeichen ihre Zustimmung, dass sie weiterhin an der aktuellen Auktion teilnehmen (offene Gebotsabgabe).⁸⁶⁸ Anschließend ruft der Auktionator einen mit der Zeit ansteigenden Preis für das Aktionsgut aus.⁸⁶⁹ Er fährt in dieser Weise so lange fort, bis nur noch ein Bieter bereit ist, den ausgerufenen Preis für das Gut zu zahlen.⁸⁷⁰ Die Anzahl der Bieter ist hierbei bekannt, die Gebote sind öffentlich.⁸⁷¹

Bei der *holländischen Auktion* ruft der Auktionator zu Anfang der Auktion einen bewusst hoch gewählten Preis aus und reduziert diesen solange, bis einer der Bieter bereit ist, das Aktionsgut zu dem ausgerufenen Preis zu kaufen (offene Gebotsabgabe).⁸⁷²

Bei der *verdeckten Erstpreisauktion* geben alle Teilnehmer ihre Gebote verdeckt ab. Den Zuschlag für das Aktionsgut erhält der Höchstbietende.⁸⁷³ Er zahlt den von ihm gebotenen Preis.⁸⁷⁴

Bei der *verdeckten Zweitpreisauktion* geben ebenfalls alle Teilnehmer ihre Gebote verdeckt ab und es erhält ebenfalls der Bieter des höchsten Gebotes den Zuschlag für das Aktionsgut.⁸⁷⁵ Er zahlt jedoch den Preis des zweithöchsten Gebotes.⁸⁷⁶ Diese Vorgehensweise bedeutet nicht zwangsläufig einen Verlust für den Verkäufer, da die Teilnehmer regelmäßig höhere Gebote als in Erstpreisauktionen abgeben.⁸⁷⁷

5.3.3 Moderne Auktionsformen

5.3.3.1 Begriff der mehrdimensionalen Auktion

BICHLER et al. beschreiben, dass Auktionen im Vergleich mit klassischen Verhandlungsformen eine höhere alloкатive Effizienz aufweisen, ihr Einsatz im zwischenbetrieblichen Handel jedoch vielfach kritisiert wurde.⁸⁷⁸ Als Grund hierfür wird die Tatsache gesehen, dass die in der Realität nötigen Verhandlungen zu komplex sind, um sie durch klassische Auktionsformen ersetzen zu können.⁸⁷⁹ Denn nur selten liegen Verhandlungen über standardisierte, leicht beschreibbare Güter vor, bei deren Verkauf der Preis der einzige

⁸⁶⁸ Vgl. KRISHNA (2009), S. 2, sowie MILGROM/WEBER (1982), S. 1104.

⁸⁶⁹ Vgl. KRISHNA (2009), S. 2, sowie MILGROM/WEBER (1982), S. 1104.

⁸⁷⁰ Vgl. KRISHNA (2009), S. 2, sowie MILGROM/WEBER (1982), S. 1089.

⁸⁷¹ Vgl. KRISHNA (2009), S. 2, sowie MILGROM/WEBER (1982), S. 1089.

⁸⁷² Vgl. MILGROM/WEBER (1982), S. 1089.

⁸⁷³ Vgl. MILGROM/WEBER (1982), S. 1090.

⁸⁷⁴ Vgl. MILGROM/WEBER (1982), S. 1090.

⁸⁷⁵ Vgl. MILGROM/WEBER (1982), S. 1090.

⁸⁷⁶ Vgl. MILGROM/WEBER (1982), S. 1090.

⁸⁷⁷ Vgl. MILGROM/WEBER (1982), S. 1090.

⁸⁷⁸ Vgl. BICHLER et al. (2005), S. 126.

⁸⁷⁹ Vgl. BICHLER et al. (2005), S. 126.

Verhandlungspunkt ist.⁸⁸⁰ Stattdessen werden regelmäßig Verhandlungen über qualitative Attribute einer beliebigen Menge von Gütern geführt.⁸⁸¹ Auktionsformen, welche kommerzielle Verhandlungen in mehreren Dimensionen abbilden, werden als *mehrdimensionale Auktionen* bezeichnet.⁸⁸² Diese Dimensionen sind bspw. Menge, Art und Qualität des versteigerten Gutes.⁸⁸³

Mehrdimensionale Auktionen basieren auf dem Austausch komplexer Präferenzinformationen, wodurch sie mit der Etablierung moderner Informations- und Kommunikationstechnologien – und den damit einhergehenden, verbesserten Möglichkeiten zum Austausch komplexer Datenstrukturen – vermehrt ins Zentrum des Interesses gerückt sind.⁸⁸⁴ Sie versprechen eine hohe alloкатive Effizienz trotz der möglicherweise komplexen Präferenzen der teilnehmenden Bieter hinsichtlich der verhandelten Dimensionen.⁸⁸⁵

Als *Mehrgutauktionen* werden Auktionen bezeichnet, bei denen mehrere Güter versteigert werden. Diese Güter können identisch, Substitute, Komplemente oder gänzlich verschieden sein.⁸⁸⁶ Sie können in mehreren Auktionen oder in einer einzigen Auktion versteigert werden.⁸⁸⁷

5.3.3.2 Begriff der kombinatorischen Auktion

Sehr gut untersuchte mehrdimensionale Auktionen zur Versteigerung mehrerer Güter sind sog. *kombinatorischen Auktionen*.⁸⁸⁸

Während die vorgestellten Standard-Auktionen auf die Preisfestsetzung bei einfachen, standardisierten Gütern beschränkt sind, ermöglichen kombinatorische Auktionen die Versteigerung von Bündeln komplexer Güter.⁸⁸⁹ So wurde bspw. von Oktober 2001 bis Juni 2002 in Nigeria erstmals eine kombinatorische Auktion zur Versteigerung der Frequenzbänder für regional beschränkte Mobilfunklizenzen durchgeführt.⁸⁹⁰ Auch in der Transportlogistik, dem innerbetrieblichen Leistungsaustausch und in der Beschaffung existieren publizierte Anwendungen.⁸⁹¹ Das Forschungsgebiet umfasst Fragen der Koordination in verteilten Systemen und gründet sich auf Erkenntnisse aus der Ökonomie, der

⁸⁸⁰ Vgl. BICHLER et al. (2005), S. 126.

⁸⁸¹ Vgl. BICHLER et al. (2005), S. 126.

⁸⁸² Vgl. BICHLER et al. (2005), S. 127.

⁸⁸³ Vgl. BICHLER et al. (2005), S. 127.

⁸⁸⁴ Vgl. BICHLER et al. (2005), S. 127.

⁸⁸⁵ Vgl. BICHLER et al. (2005), S. 127.

⁸⁸⁶ Vgl. KRISHNA (2009), S. 173.

⁸⁸⁷ Vgl. KRISHNA (2009), S. 173.

⁸⁸⁸ Vgl. BICHLER et al. (2005), S. 127.

⁸⁸⁹ Vgl. BICHLER et al. (2005), S. 126.

⁸⁹⁰ Vgl. KOBOLDT et al. (2003), S. 3.

⁸⁹¹ Vgl. PERENNES (2013), ACKERMANN et al. (2011), GUJO et al. (2007), sowie BICHLER et al. (2005).

künstlichen Intelligenz, der Informatik, sowie des Operations Research.⁸⁹² Da mehrdimensionale Auktionen der effizienten Durchführung betrieblicher Prozesse dienen, bedürfen sie der Unterstützung durch spezialisierte Informations- und Kommunikationstechnologien.⁸⁹³

Eine kombinatorische Auktion ermöglicht es den teilnehmenden Bietern, Gebote auf Bündel von Gütern abzugeben.⁸⁹⁴ Abbildung 14 zeigt eine solche Auktion.

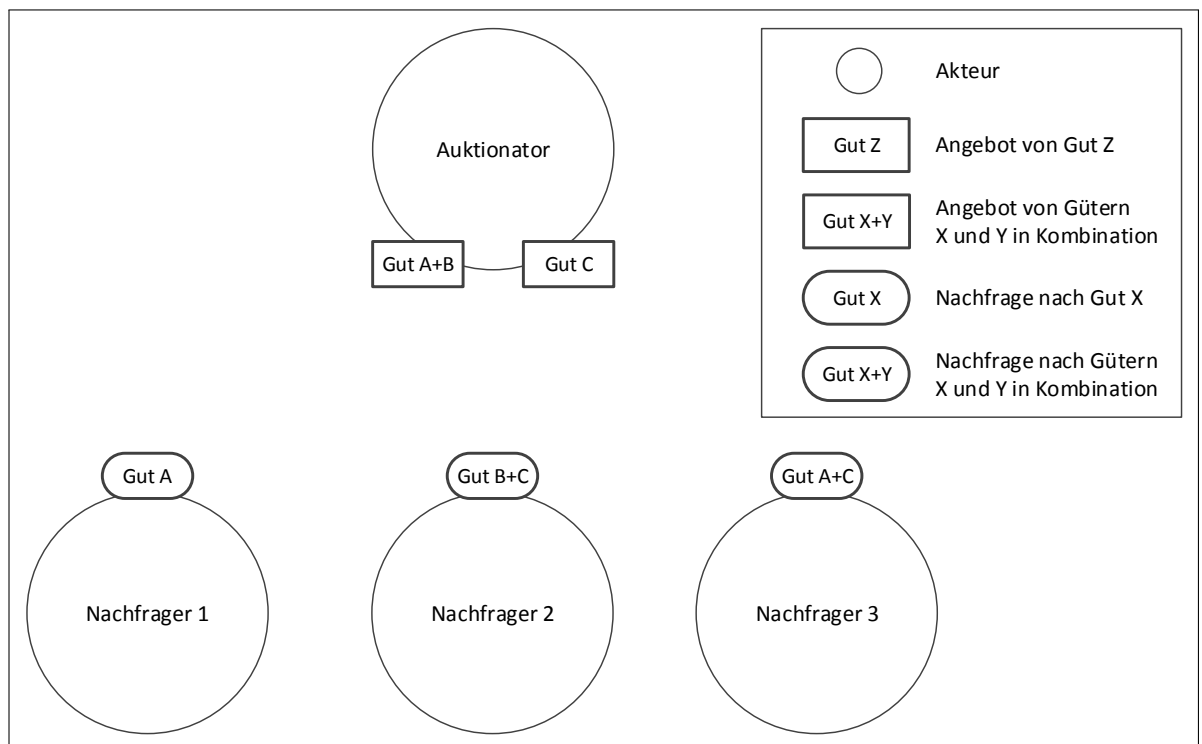


Abbildung 14: Grafisches Beispiel für eine einseitige kombinatorische Auktion⁸⁹⁵

Abbildung 14 zeigt ein Beispiel, in welchem ein Auktionator drei Güter (A , B und C) versteigert. Zwei dieser Güter (A und B) werden als Güterbündel angeboten, d.h., in der Endallokation müssen beide Güter vorkommen. Es kann nicht nur Gut A oder nur Gut B versteigert werden. Nachfrager 1 und 2 können gemeinsam den Zuschlag erhalten, da ihre Gebote sich in einer Weise ergänzen, dass alle Bündel versteigert werden können (das Bündel A und B wird zerlegt, um die Gebote von Nachfrager 1 und 2 zu erfüllen). Nachfrager 3 wird dagegen in keinem Fall den Zuschlag erhalten, da sein Gebot eine Teilung des Güterbündel $A + B$ erfordern würde (da kein Nachfrager Gut B einzeln nachfragt).

⁸⁹² Vgl. BICHLER et al. (2005), S. 127.

⁸⁹³ Vgl. BICHLER et al. (2005), S. 127.

⁸⁹⁴ Vgl. BICHLER et al. (2005), S. 127.

⁸⁹⁵ Quelle: eigene Darstellung.

Die Wahl dieser Auktionsform erscheint immer dann sinnvoll, wenn die versteigerten Güter Komplemente darstellen.⁸⁹⁶ Aus Sicht der Bieter wäre es in diesem Fall nicht effizient zu versuchen, die betreffenden Güter in separaten Auktionen zu erstehen, da sie sonst Gefahr laufen würden, nur einen Teil der von ihnen präferierten Allokation zu erhalten.⁸⁹⁷ Aus diesem Grund lässt sich die Präferenz für eine bestimmte Allokation in kombinatorischen Auktionen durch Bündelgebote ausdrücken.⁸⁹⁸ Dies bedeutet, dass der gebotene Preis nur für das Bündel gültig ist und nur gezahlt wird, wenn der Bieter den Zuschlag für das gesamte Bündel erhält.⁸⁹⁹ Das Gebot ist nicht teilbar.⁹⁰⁰

5.3.3.3 Ziele des Einsatzes kombinatorischer Auktionen

BICHLER et al. haben durch die Befragung von Softwareherstellern fünf Ziele des Einsatzes kombinatorischer Auktionen in Unternehmen identifiziert:⁹⁰¹

1. **Kosteneinsparungen:** Teilweise wurde das Einsparungspotential durch den Einsatz kombinatorischer Auktionen auf bis zu 13% beziffert, wobei Einsparungen auch durch Prozessverbesserungen entstehen, welche mit dem Einsatz von kombinatorischen Auktionen einhergehen.⁹⁰²
2. **Senkung der Transaktionskosten zur Durchführung komplexer Verhandlungen:** Die Alternative zum Einsatz kombinatorischer Auktionen ist in der Regel die Durchführung parallel stattfindender, bilateraler Verhandlungen.⁹⁰³ Im Gegensatz hierzu bieten kombinatorische Auktionen eine Kosten und Zeit sparende Möglichkeit der effektiven Durchführung komplexer Verhandlungen über mehrere Güter.⁹⁰⁴
3. **Transparenz:** Die Markttransparenz wird durch die Durchführung offener, iterativer Auktionen erhöht.⁹⁰⁵
4. **Fairness:** Alle Teilnehmer einer Auktion werden nach den gleichen Regeln behandelt, was in einigen Anwendungsfällen, wie bspw. der öffentlichen Beschaffung, eine hohe Priorität haben kann.⁹⁰⁶

⁸⁹⁶ Vgl. BICHLER et al. (2005), S. 127.

⁸⁹⁷ Vgl. BICHLER et al. (2005), S. 127.

⁸⁹⁸ Vgl. BICHLER et al. (2005), S. 127.

⁸⁹⁹ Vgl. BICHLER et al. (2005), S. 127.

⁹⁰⁰ Vgl. BICHLER et al. (2005), S. 127.

⁹⁰¹ Vgl. BICHLER et al. (2005), S. 132.

⁹⁰² Vgl. BICHLER et al. (2005), S. 132.

⁹⁰³ Vgl. BICHLER et al. (2005), S. 132.

⁹⁰⁴ Vgl. BICHLER et al. (2005), S. 132.

⁹⁰⁵ Vgl. BICHLER et al. (2005), S. 133.

⁹⁰⁶ Vgl. BICHLER et al. (2005), S. 133.

5.3.3.4 Probleme von kombinatorischen Auktionen

Die Fachliteratur zu klassischen Auktionsformen befasst sich mit Untersuchungen des Auktionserlöses und der Identifizierung optimaler Bietstrategien für verschiedene Auktionsformen sowie deren strategischer Komplexität.⁹⁰⁷ Für kombinatorische Auktionen existieren nur wenige solcher Untersuchungen, was damit begründet werden kann, dass kombinatorische Auktionen ein noch vergleichsweise junges Forschungsfeld darstellen und wesentlich komplexer sind als klassische Auktionsformen.⁹⁰⁸

Ein wichtiger Aspekt bei der Gestaltung und Auswahl kombinatorischer oder generell mehrdimensionaler Auktionen ist Komplexität.⁹⁰⁹

Den teilnehmenden Bietern stellt sich das Problem der *Bewertungskomplexität*, da sie alle möglichen Güterbündel bewerten müssen, um anschließend zu entscheiden für welche der Bündel sie Gebote abgeben werden.⁹¹⁰ Für eine Auktion mit 10 Gütern sind dies bereits $2^{10} - 1 = 1023$ zu bewertende Bündel.⁹¹¹

Weiterhin ergibt sich das Problem der *strategischen Komplexität* beim Festlegen einer optimalen Bietstrategie, bei der jeder teilnehmende Bieter entscheiden muss, für welche der möglichen Güterbündel er Gebote abgeben möchte.⁹¹²

Dem Auktionator stellen sich Probleme durch die *Zeitkomplexität des Allokationsproblems* und aus der Berechnung von Preisen und Zahlungen.⁹¹³ Allen Teilnehmern stellt sich das Problem der *Kommunikationskomplexität*, da die Anzahl der Nachrichten, welche zwischen Auktionator und teilnehmenden Bietern ausgetauscht werden müssen, im ungünstigsten Fall exponentiell mit der Anzahl der versteigerten Güter steigt.⁹¹⁴

Wie auch bei den vorgestellten Standard-Auktionen können kombinatorische Auktionsmodelle in geschlossene und offene Auktionsmodelle unterschieden werden.⁹¹⁵ Zu den verbreiteten geschlossenen Auktionsmodellen zählen *geschlossene Bestpreisauktionen*.⁹¹⁶

Geschlossene Bestpreisauktionen sind eine Anwendung kombinatorischer Auktionen, bei der alle Gebote verschlossen bis zu einem bestimmten Termin abgegeben werden und der

⁹⁰⁷ Vgl. BICHLER et al. (2005), S. 128.

⁹⁰⁸ Vgl. BICHLER et al. (2005), S. 128.

⁹⁰⁹ Vgl. BICHLER et al. (2005), S. 129.

⁹¹⁰ Vgl. BICHLER et al. (2005), S. 129.

⁹¹¹ Vgl. BICHLER et al. (2005), S. 129.

⁹¹² Vgl. BICHLER et al. (2005), S. 129.

⁹¹³ Vgl. BICHLER et al. (2005), S. 129.

⁹¹⁴ Ähnliche Probleme können auch bei multivariaten Auktionen und Mengenrabattauktionen beobachtet werden. Vgl. BICHLER et al. (2005), S. 129.

⁹¹⁵ Vgl. BICHLER et al. (2005), S. 130.

⁹¹⁶ Vgl. BICHLER et al. (2005), S. 130.

Auktionator anschließend die ertragsmaximierende Kombination an Geboten wählt.⁹¹⁷ Die strategische Komplexität ist bei Bestpreisauktionen im Allgemeinen hoch, dafür sind sie stabil gegen Absprachen.⁹¹⁸

In offenen Auktionsmodellen erhalten die teilnehmenden Bieter Informationen übereinander, welche dazu führen können, dass sie ihre eigenen Gebote ändern.⁹¹⁹ Hierzu muss die offene Auktion als iterative Auktion über mehrere Gebotsrunden durchgeführt werden.⁹²⁰

Argumente für den Einsatz iterativer Auktionsmodelle sind:

Im Gegensatz zu geschlossenen kombinatorischen Auktionen müssen bei offenen kombinatorischen Auktionen nicht alle $2^m - 1$ Güterbündel bewertet werden, da es reicht für einige Güterbündel Gebote abzugeben.⁹²¹ So haben die teilnehmenden Bieter die Möglichkeit, ihre Präferenzen an neue Informationen anzupassen und in späteren Runden Gebote auf Bündel abzugeben, die sie zu Anfang der Auktion nicht in Betracht gezogen hatten.⁹²²

Es werden Auktionsmodelle gesucht, welche die Komplexität für Bieter und Auktionator minimieren, ohne auf wichtige ökonomische Eigenschaften wie alloкатive Effizienz, Strategiebeständigkeit, budgetäre Ausgeglichenheit und individuelle Rationalität verzichten zu müssen.⁹²³

Argumente gegen den Einsatz iterativer Auktionsmodelle sind hingegen:

Das *Schwellwertproblem* beschreibt die Schwierigkeit, sich als teilnehmender Bieter gegen ein großes Bündelgebot durchzusetzen.⁹²⁴

Das folgende Beispiel illustriert diesen Sachverhalt:⁹²⁵

Es werden drei Güter A, B und C verkauft. Bieter 1, 2 und 3 sind bereit, jeweils eines der Güter für 4 EUR zu kaufen. Bieter 1, 2 und 3 haben in der aktuellen Runde bereits jeweils 3 EUR geboten. Bieter 4 bietet für das gesamte Bündel 10 EUR. Die Bieter 1, 2 und 3 sind mit dem Schwellwertproblem konfrontiert: Keiner von ihnen kann das Gewinnergebot von Bieter 4 alleine überbieten. Einer von ihnen müsste sein Gebot um mehr

⁹¹⁷ Vgl. BICHLER et al. (2005), S. 130.

⁹¹⁸ Vgl. BICHLER et al. (2005), S. 130.

⁹¹⁹ Vgl. BICHLER et al. (2005), S. 131.

⁹²⁰ Vgl. BICHLER et al. (2005), S. 131.

⁹²¹ Vgl. BICHLER et al. (2005), S. 131.

⁹²² Vgl. BICHLER et al. (2005), S. 131.

⁹²³ Vgl. BICHLER et al. (2005), S. 131.

⁹²⁴ Vgl. BICHLER et al. (2005), S. 131.

⁹²⁵ Das Beispiel ist an das Beispiel einer Beschaffungsauktion von BICHLER et al. angelehnt. Vgl. BICHLER et al. (2005), S. 131.

als 1 EUR steigern und so mehr als seine private Bewertung bieten. Das Schwellwertproblem würde somit eine Koordination zwischen den Bieter 1, 2 und 3 erforderlich machen, was in der Praxis oft schwer zu bewerkstelligen ist.⁹²⁶ Hierdurch laufen iterative Auktionen Gefahr, eine allokativ ineffiziente und damit auch nicht ertragsmaximale Lösung zu erreichen.⁹²⁷

Das *Bloßstellungsproblem* ist primär im Zusammenhang mit simultan durchgeführten Auktionen relevant.⁹²⁸ Werden zwei Güter versteigert und ein Bieter möchte beide und nicht nur eines der beiden Güter ersteigern, so läuft er in simultanen Auktionen Gefahr nur eines der Güter zu ersteigern.⁹²⁹ Bieter verhalten sich in solchen Situation mit ihren Geboten eher zurückhaltend, was zu ineffizienten Allokationen führen kann.⁹³⁰

Zwar wird dieses Problem durch den Einsatz von Bündelgeboten weitgehend gelöst, jedoch können ähnliche Probleme auch bei kombinatorischen Auktionen auftreten.⁹³¹

So ist eine Situation denkbar, in der ein Bieter nur eines von zwei versteigerten Gütern kaufen möchte, aber nicht beide.⁹³² Besonders in iterativen Auktionen führt dieses Problem zu Schwierigkeiten.⁹³³ Eine mögliche Lösung ist das Zulassen des Zurückziehens von Geboten der vorangegangenen Runde während der aktuellen Runde.⁹³⁴ Eine andere Lösung wäre die Implementierung einer *Bietsprache*, welche eine logische Verknüpfung von Geboten zulässt.⁹³⁵ Allerdings würden hierdurch auch die Komplexität des Allokationsproblems und die Anforderungen an die Benutzerschnittstelle der Software steigen.⁹³⁶

Das *Problem der gleichwertigen Allokationen* besteht bei der Abgabe gleichwertiger Gebote auf die gleiche Kombination an Gütern.⁹³⁷ In klassischen Auktionen wird hier regelmäßig das Gebot gewählt, welches als erstes abgegeben wurde.⁹³⁸ In einer kombinatorischen Auktion besteht eine Allokation oft aus mehreren Geboten und während einer Auktionsrunde können mehrere unterschiedliche, jedoch gleich gute Allokationen zustan-

⁹²⁶ Vgl. BICHLER et al. (2005), S. 131.

⁹²⁷ Vgl. BICHLER et al. (2005), S. 131.

⁹²⁸ Vgl. BICHLER et al. (2005), S. 131.

⁹²⁹ Vgl. BICHLER et al. (2005), S. 131.

⁹³⁰ Vgl. BICHLER et al. (2005), S. 131.

⁹³¹ Vgl. BICHLER et al. (2005), S. 131.

⁹³² Vgl. BICHLER et al. (2005), S. 131.

⁹³³ Vgl. BICHLER et al. (2005), S. 131.

⁹³⁴ Vgl. BICHLER et al. (2005), S. 131.

⁹³⁵ Vgl. BICHLER et al. (2005), S. 131.

⁹³⁶ Vgl. BICHLER et al. (2005), S. 131.

⁹³⁷ Vgl. BICHLER et al. (2005), S. 131.

⁹³⁸ Vgl. BICHLER et al. (2005), S. 131.

de kommen.⁹³⁹ Als Lösung wird die Verwendung von Zeitstempeln diskutiert, welche es ermöglichen würden zu determinieren, welche Allokation als erste zustande gekommen ist.⁹⁴⁰

Das *Problem der Berechnung von Gleichgewichtspreisen* beschreibt das Problem der teilnehmenden Bieter, sich bei ihren Geboten an Marktpreisen zu orientieren.⁹⁴¹ In einer klassischen englischen Auktion stellt das Gebot des Höchstbietenden den aktuellen Marktpreis des Gutes dar.⁹⁴² Bieter müssen diesen Preis überbieten, um die Auktion zu gewinnen.⁹⁴³ In iterativen kombinatorischen Auktionen wird die Feststellung solcher Preise pro Gut erschwert, da eine grundlegende Anforderung an Marktpreise darin besteht, dass diese höher sein müssen als die Preise von Verlierergeboten aber nicht höher als die Preise der Gewinner der aktuellen Runde.⁹⁴⁴ Zudem darf es keine bessere Allokation für die gegebenen Marktpreise geben, welche aus diesem Grund auch *Gleichgewichtspreise* genannt werden.⁹⁴⁵ Im Falle linearer Preise, welche den Gleichgewichtspreis eines Güterbündels als Summe der Einzelpreise der enthaltenen Güter definieren, ist dies leicht verständlich.⁹⁴⁶ In kombinatorischen Auktionen lassen sich jedoch regelmäßig keine linearen Preise berechnen, teilweise ist sogar die Berechnung anonymer nicht-linearer Preise unmöglich und es müssen personalisierte nicht-lineare Preise berechnet werden.⁹⁴⁷ Hierbei werden unterschiedlichen Bietern unterschiedliche Preise für dieselben Güterbündel zugeordnet, da der Gleichgewichtspreis von den privaten Bewertungen der einzelnen Bieter ab.⁹⁴⁸

Viele Implementierungen kombinatorischer Auktionen lassen die optimale Allokation zentral vom Auktionator berechnen.⁹⁴⁹

Auf Basis dieser Betrachtungen soll im Folgenden das Modell einer nicht iterativen, geschlossenen Bestpreisauktion betrachtet werden. Dabei muss beachtet werden, ob eine Möglichkeit besteht, die strategische Komplexität und Bewertungskomplexität für die Teilnehmer durch eine gezielte Gestaltung der Anbahnung von Auktionen möglichst gering zu halten.

⁹³⁹ Vgl. BICHLER et al. (2005), S. 131.

⁹⁴⁰ Vgl. BICHLER et al. (2005), S. 131.

⁹⁴¹ Vgl. BICHLER et al. (2005), S. 131.

⁹⁴² Vgl. BICHLER et al. (2005), S. 131.

⁹⁴³ Vgl. BICHLER et al. (2005), S. 131.

⁹⁴⁴ Vgl. BICHLER et al. (2005), S. 131.

⁹⁴⁵ Vgl. BICHLER et al. (2005), S. 131.

⁹⁴⁶ Vgl. BICHLER et al. (2005), S. 131.

⁹⁴⁷ Vgl. BICHLER et al. (2005), S. 131.

⁹⁴⁸ Vgl. BICHLER et al. (2005), S. 131.

⁹⁴⁹ Vgl. BICHLER et al. (2005), S. 132.

5.3.3.5 Modell einer kombinatorischen Auktion

Aufgrund der Zeitkomplexität des Allokationsproblems und der strategischen Komplexität für die Bieter wurden kombinatorische Auktionen noch Mitte der 1990er Jahre als undurchführbar eingeschätzt.⁹⁵⁰ Mittlerweile lässt sich das Allokationsproblem für viele in der Praxis vorkommende Problemgrößen in annehmbarer Zeit lösen.⁹⁵¹ Das relevantere Problem stellt die strategische Komplexität für die Bieter dar.⁹⁵² Alle Bieter müssen zur Festlegung ihrer Bietstrategie – zumindest theoretisch – ihre individuellen Präferenzen für alle möglichen Güterbündel berechnen.⁹⁵³ Sollten die Bieter falsche oder unvollständige Berechnungen anstellen, so kann dies theoretisch zu einer ineffizienten Allokation im Ergebnis der Auktion führen.⁹⁵⁴

Das resultierende Allokationsproblem lässt sich als binäres Programm formulieren:⁹⁵⁵

Eine Menge M von Gütern k mit $k = \{1, 2, \dots, m\}$ (also $M = \{1, 2, \dots, m\}$) wird im Rahmen einer Menge aus Geboten $Y = \{B_1, B_2, \dots, B_n\}$ versteigert. Der Auktionator verfügt über eine zuvor definierte Quantität jedes Gutes k : $U = \{u_1, u_2, \dots, u_m\}$. Ein Gebot ist ein Tupel $B_j = \{\{\lambda_j^1, \lambda_j^2, \dots, \lambda_j^m\}, p_j\}$. λ_j^k bezeichnet die nachgefragte Quantität des Gutes k . Der Preis des Gebots ist durch p_j dargestellt. Die Entscheidungsvariable x_j markiert die Gebote B_j nach „Zuschlag“ ($x_j = 1$) und „kein Zuschlag“ ($x_j = 0$).

Das Optimierungsproblem kann nun entsprechend formuliert werden:⁹⁵⁶

$$\begin{aligned}
 \max \quad & \sum_{j=1}^n p_j x_j \\
 \text{s.t.} \quad & \sum_{j=1}^n \lambda_j^k x_j = u_k \quad k = 1, 2, \dots, m \\
 & x_j \in \{0, 1\} \quad j = 1, 2, \dots, n \\
 & \lambda_j^k, p_j, u_k \in \mathbb{R}_{\geq 0} \quad j = 1, 2, \dots, n \\
 & \quad \quad \quad k = 1, 2, \dots, m
 \end{aligned}$$

Das resultierende Allokationsproblem ist im Gegensatz zu klassischen Auktionsmodellen NP-vollständig, d.h. die Zeitkomplexität steigt bei steigender Güteranzahl m oder Bieteranzahl n exponentiell.⁹⁵⁷

⁹⁵⁰ Vgl. BICHLER et al. (2005), S. 127.

⁹⁵¹ Vgl. BICHLER et al. (2005), S. 127.

⁹⁵² Vgl. BICHLER et al. (2005), S. 127.

⁹⁵³ Vgl. BICHLER et al. (2005), S. 127.

⁹⁵⁴ Vgl. BICHLER et al. (2005), S. 127.

⁹⁵⁵ Vgl. SANDHOLM et al. (2002), S. 70. Ähnliche Ausführungen finden sich bei BICHLER et al.. Vgl. BICHLER et al. (2005), S. 128.

⁹⁵⁶ Vgl. SANDHOLM et al. (2002), S. 70.

⁹⁵⁷ Vgl. BICHLER et al. (2005), S. 128.

Die in der Praxis auftretenden Problemgrößen haben sich jedoch als handhabbar herausgestellt.⁹⁵⁸ Auch Auktionen mit hunderten Geboten auf mehrere Dutzend Güter konnten in Sekunden gelöst werden.⁹⁵⁹ Für sehr große Auktionen wurden der Einsatz von exakten Lösungsverfahren oder Meta-Heuristiken diskutiert.⁹⁶⁰ Der Einsatz von heuristischen Verfahren zur Lösung des Allokationsproblems ist jedoch problematisch, da die mathematische Optimalität der Problemlösung für die ökonomische Effizienz der Allokation von hoher Bedeutung ist und die Auswahl der mathematisch gesehen zweit- oder drittbesten Allokation eine abweichende Güterverteilung bedeuten würde.⁹⁶¹

5.3.3.6 Modell einer Börse

Eine zweiseitige kombinatorische Auktion mit mehreren Anbietern und mehreren Nachfragern stellt eine Börse dar.⁹⁶² Die Teilnehmer einer Börse sind dazu berechtigt zu kaufen und zu verkaufen, nur zu kaufen oder nur zu verkaufen.⁹⁶³ Hieraus ergibt sich, dass alle Standard-Auktionen ein Spezialfall der Auktionsform Börse sind.⁹⁶⁴

Der Administrator der Börse determiniert eine Menge M von Gütern k mit $k = \{1, 2, \dots, m\}$ (also $M = \{1, 2, \dots, m\}$), welche von den insgesamt n Anbietern und Nachfragern in der Börse gehandelt werden.⁹⁶⁵ Lediglich diese Güter dürfen in den Angeboten und Nachfragen berücksichtigt werden.⁹⁶⁶ Ein Gebot ist definiert als $B_j = ((\lambda_j^1, \lambda_j^2, \dots, \lambda_j^m), p_j)$, wobei λ_j^k die nachgefragte oder angebotene Quantität eines Gutes und p_j der Preis des Bündels j ist.⁹⁶⁷ Eine positive Quantität λ_j^k steht für eine Nachfrage nach, ein negativer Wert steht für ein Angebot von Gut k .⁹⁶⁸

Ein positiver Preis p_j steht für eine Nachfrage nach Bündel j .⁹⁶⁹ Der Preis beschreibt die maximale Zahlungsbereitschaft für das Bündel.

Ein negativer Preis p_j steht für ein Angebot von Bündel j .⁹⁷⁰ Der Betrag des Preises beschreibt die minimale Zahlungsforderung für das Bündel.

⁹⁵⁸ Vgl. BICHLER et al. (2005), S. 128.

⁹⁵⁹ Vgl. BICHLER et al. (2005), S. 128.

⁹⁶⁰ Vgl. BICHLER et al. (2005), S. 128.

⁹⁶¹ Vgl. BICHLER et al. (2005), S. 128.

⁹⁶² Vgl. SANDHOLM et al. (2002), S. 71.

⁹⁶³ Vgl. SANDHOLM et al. (2002), S. 71.

⁹⁶⁴ Vgl. SANDHOLM et al. (2002), S. 71.

⁹⁶⁵ Vgl. SANDHOLM et al. (2002), S. 71.

⁹⁶⁶ Vgl. SANDHOLM et al. (2002), S. 71.

⁹⁶⁷ Vgl. SANDHOLM et al. (2002), S. 71.

⁹⁶⁸ Vgl. SANDHOLM et al. (2002), S. 71 f.

⁹⁶⁹ Im englischen Original als „bid“ (Kaufgesuch) bezeichnet. Vgl. SANDHOLM et al. (2002), S. 72.

⁹⁷⁰ Im englischen Original als „ask“ (Preisangebot) bezeichnet. Vgl. SANDHOLM et al. (2002), S. 72.

Das Modell markiert die Angebote und Nachfragen nach „Zuschlag“ ($x_j = 1$) und „kein Zuschlag“ ($x_j = 0$), um die *Differenz zwischen der gesamten maximalen Zahlungsbereitschaft und der gesamten minimalen Zahlungsforderung* unter der Prämisse zu maximieren, dass die Nachfrage das Angebot nicht übersteigt.⁹⁷¹ In diesem Modell ist die nachgefragte bzw. angebotene Quantität jedes Gutes stets 0 oder 1, d.h. es wird entweder eine Indifferenz gegenüber dem Gut k geäußert ($\lambda_j^k = 0$) oder genau ein Gut nachgefragt ($\lambda_j^k = 1$) bzw. genau ein Gut angeboten ($\lambda_j^k = -1$).⁹⁷²

$$\begin{aligned}
 & \max \quad \sum_{j=1}^n p_j x_j \\
 & \text{s.t.} \quad \sum_{j=1}^n \lambda_j^k x_j = 0 \quad k = 1, 2, \dots, m \\
 & \quad p_j \in \mathbb{R} \quad j = 1, 2, \dots, n \\
 & \quad x_j \in \{0, 1\} \quad j = 1, 2, \dots, n \\
 & \quad \lambda_j^k \in \{-1, 0, 1\} \quad j = 1, 2, \dots, n \\
 & \quad \quad \quad k = 1, 2, \dots, m
 \end{aligned}$$

Die durch die erste Zeile beschriebene Differenz zwischen der gesamten maximalen Zahlungsbereitschaft und der gesamten minimalen Zahlungsforderung wird aus Gründen der Lesbarkeit im Folgenden als *Auktionsmehrwert* bezeichnet.

Ein Beispiel soll das Modell und den Auktionsmehrwert illustrieren:

- **Gebot 1**

Anbieter 1 verkauft Gut A und B in Kombination für 30 EUR.

- **Gebot 2**

Anbieter 2 verkauft Gut C für 50 EUR.

- **Gebot 3**

Nachfrager 1 würde Gut A für 30 EUR kaufen.

- **Gebot 4**

Nachfrager 2 würde die Güter B und C in Kombination für 60 EUR kaufen.

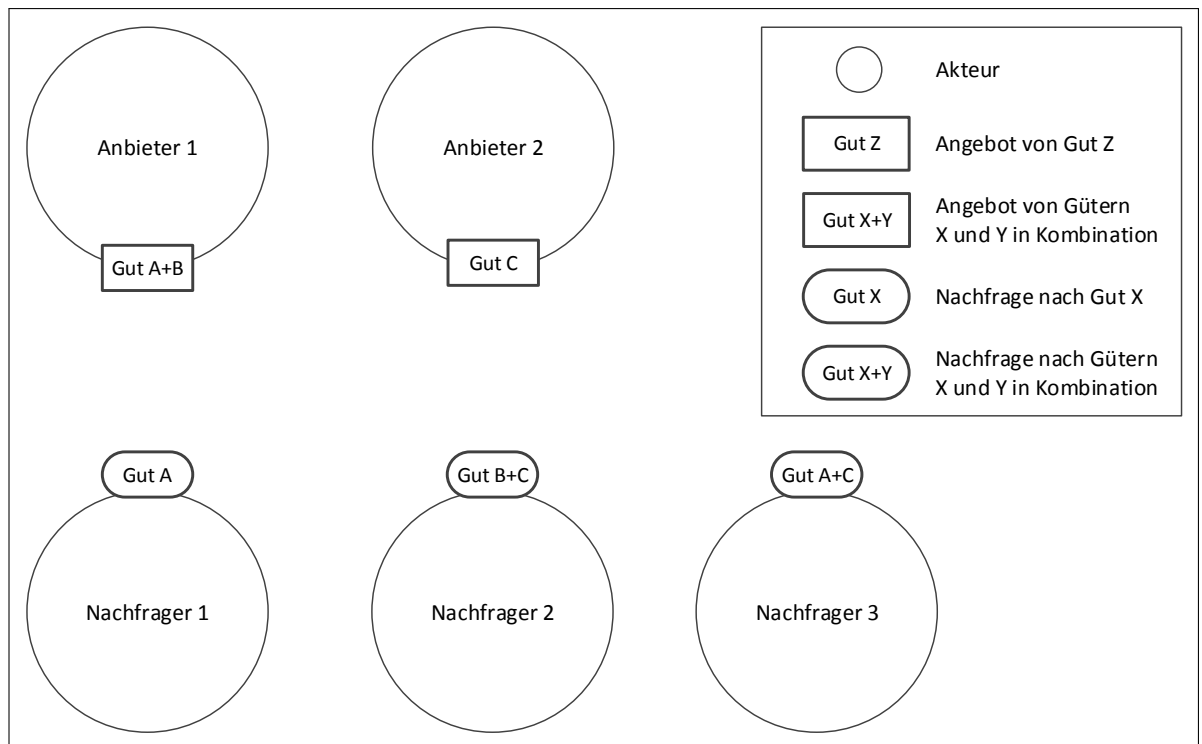
- **Gebot 5**

Nachfrager 3 würde die Güter A und C in Kombination für 100 EUR kaufen.

Abbildung 15 zeigt eine solche Auktion.

⁹⁷¹ Vgl. SANDHOLM et al. (2002), S. 72.

⁹⁷² Der Aspekt, dass stets nur ein Gut angeboten oder nachgefragt wird, wird in der späteren Implementierung relevant, da individuelle digitale Beschreibungen der Transportdienstleistungen gehandelt werden. Vgl. Kapitel 7.4.2.2, S. 217.

Abbildung 15: Grafisches Beispiel für eine Börse⁹⁷³**Szenario 1**

	λ_j^A	λ_j^B	λ_j^C	p_j	x_j
Gebot 1	-1	-1	0	-30	1
Gebot 2	0	0	-1	-50	1
Gebot 3	1	0	0	30	1
Gebot 4	0	1	1	60	1
Gebot 5	1	0	1	100	0

Tabelle 3: Zahlenbeispiel für eine Börse⁹⁷⁴

Der Auktionsmehrwert ist: $(-30) \cdot 1 + (-50) \cdot 1 + 30 \cdot 1 + 60 \cdot 1 + 100 \cdot 0 = 10$

Die Nebenbindungen bzgl. der Gütermengen sind erfüllt:

Für Gut A: $(-1) \cdot 1 + 0 \cdot 1 + 1 \cdot 1 + 0 \cdot 1 + 1 \cdot 0 = 0$

Für Gut B: $(-1) \cdot 1 + 0 \cdot 1 + 0 \cdot 1 + 1 \cdot 1 + 0 \cdot 0 = 0$

Für Gut C: $0 \cdot 1 + (-1) \cdot 1 + 0 \cdot 1 + 1 \cdot 1 + 1 \cdot 0 = 0$

Daher erhalten die Gebote 1, 2, 3 und 4 den Zuschlag.

⁹⁷³ Quelle: eigene Darstellung.

⁹⁷⁴ Quelle: eigene Darstellung.

Szenario 2

	λ_j^A	λ_j^B	λ_j^C	p_j	x_j
Gebot 1	-1	-1	0	-30	1
Gebot 2	0	0	-1	-50	1
Gebot 3	1	0	0	30	0
Gebot 4	0	1	1	60	0
Gebot 5	1	0	1	100	1

Tabelle 4: Zahlenbeispiel für ein invalides Szenario⁹⁷⁵

Der Auktionsmehrwert ist: $(-30)*1 + (-50)*1 + 30*0 + 60*0 + 100*1 = 20$

Er ist größer als in Szenario 1, jedoch sind die Nebenbindungen bzgl. der Gütermengen nicht erfüllt:

Für Gut A: $(-1) * 1 + 0 * 1 + 1 * 0 + 0 * 0 + 1 * 1 = 0$

Für Gut B: $(-1) * 1 + 0 * 1 + 0 * 0 + 1 * 0 + 0 * 1 = -1$

Für Gut C: $0 * 1 + (-1) * 1 + 0 * 0 + 1 * 0 + 1 * 1 = 0$

Daher können die Gebote 1, 2 und 5 nicht den Zuschlag erhalten.

Dies offenbart ein Gestaltungsproblem: Es sind leicht Szenarien denkbar, in denen Nachfrager 3 indifferent wäre, ob die ihm zugeschlagene Endallokation Gut C enthält oder nicht, solange er das von ihm nachgefragte Güterbündel erhält. Dies wäre insbesondere denkbar, wenn die Ersteigerung zusätzlicher, nicht nachgefragter Güter keine zusätzlichen Kosten für den Gewinner bedeutet, da bei ihrer Nichtnutzung bspw. keine Kosten anfallen. Dies wäre bei immateriellen Gütern wie bspw. Nutzungsrechten der Fall. Dieses Problem der sog. „kostenfreien Entsorgung“⁹⁷⁶ wird von SANDHOLM et al. vor dem Hintergrund verschiedener Auktionsmodelle diskutiert.⁹⁷⁷

⁹⁷⁵ Quelle: eigene Darstellung.

⁹⁷⁶ Im Englischen „free disposal“. Gemeint ist, dass der Auktionator bereit ist, Güter, für die zwar von einem Nachfrager nicht geboten wurde, die aber in der ermittelten Güterallokation diesem Nachfrager zugeordnet werden, zu übernehmen.
Vgl. SANDHOLM et al. (2002), S. 71.

⁹⁷⁷ Vgl. SANDHOLM et al. (2002), S. 72 f.

5.3.4 Relevanz für die wissenschaftlichen Probleme

Die Grundlagen zu Auktionen zeigen, welche Auktionsmodelle für die Konzeption eines prototypischen Systems autonomer, dezentral organisierter Software-Agenten zur Durchführung automatisierter Auktionen von multimodalen Transportdienstleistungen in Frage kommen: Zweiseitige Auktionen und multidimensionale Auktionsmodelle bieten die Möglichkeit, sowohl mehrere Anbieter als auch mehrere Nachfrager miteinander zu koordinieren sowie simultan über verschiedene Kontrakteigenschaften zu verhandeln.⁹⁷⁸

Wie auf vielen anderen Märkten⁹⁷⁹ auch, herrschen im Logistikmarkt polypolistische, multidimensionale Rahmenbedingungen vor, da sich mehrere Anbieter und mehrere Nachfrager von Transportdienstleistungen hinsichtlich mehrerer Kontrakteigenschaften wie Preis, Frachtkapazität und Transportgeschwindigkeit koordinieren müssen.⁹⁸⁰

Zur Konzipierung eines dezentralen Marktplatzes, in welchem alle Teilnehmer kontextabhängig sowohl als Anbieter wie auch als Nachfrager für multimodale Transportdienstleistungen auftreten können, ist der Einsatz von zweiseitigen Multi-Objekt-Auktionsmodellen erforderlich. Dieser Marktplatz ist aus auktionstheoretischer Sicht als Börse für Kombinationen von Transportdienstleistungen zu sehen.

5.4 Transfer der Grundlagen

Die Zusammenführung von Erkenntnissen über elektronische Marktplätze, kombinatorische Auktionen und Multi-Agenten-Systeme steht im Zentrum dieser Forschungsarbeit. Die Konzeption und Implementierung von Agenten, welche automatisiert Auktionen durchführen, ist jedoch kein neuer Gedanke: Einsatzgebiete derartiger Technologien umfassen Wertpapierbörsen, Logistikmärkte sowie die innerbetriebliche Ressourcenallokation.⁹⁸¹

PETERS stellt fest, dass die vollständige Automatisierung der Verhandlungen für die Kosten- und Zeiteffizienz eines Marktplatzes von Bedeutung ist, und entwirft ein allgemein verwendbares Marktsystem, das auf polypolistische Marktszenarien mit multilateralen Verhandlungen und einen differenzierten Interessenausgleich zwischen Anbietern und Nachfragern hinsichtlich der Transaktionspreise und weiterer Kontrakteigenschaften abzielt.⁹⁸²

⁹⁷⁸ Vgl. PETERS (2002), S. 132.

⁹⁷⁹ Weitere Beispiele sind Rohstoff-, Elektrizitäts- und Telekommunikationsmärkte. Vgl. PETERS (2002), S. 132.

⁹⁸⁰ Vgl. PETERS (2002), S. 132.

⁹⁸¹ Vgl. GUJO et al. (2007), BICHLER et al. (2005), DAVIDSSON et al. (2005), sowie PETERS (2002).

⁹⁸² Vgl. PETERS (2002), S. 131.

Viele in der Realität vorhandene elektronische Marktplätze beschränken sich auf die Vermittlung zwischen einem Anbieter und vielen Nachfragern und sind auf die Verhandlung des Preises als einziges Kriterium des Interessenausgleichs beschränkt.⁹⁸³

Das Internet und die agentenorientierte Gestaltung von Softwaresystemen stellen hierbei die Grundlage zur Verknüpfung von Erkenntnissen aus Spieltheorie und Informatik dar, welche den Einsatz mehrdimensionaler Auktionsmodelle ermöglichen.⁹⁸⁴ Hierdurch werden Automatisierungskonzepte in der Praxis möglich, welche ohne diese Technologien nicht realisierbar wären, und es entsteht die Möglichkeit eines interdisziplinären Ansatzes zur Konzeption elektronischer Marktplätze.⁹⁸⁵

Wird zusätzlich die Auktionsfrequenz der verwendeten Software-Agenten hoch genug gewählt, so ergibt sich in automatisierten Auktionen näherungsweise ein kontinuierlicher Handel.⁹⁸⁶ Die erreichbare Auktionsfrequenz ist hierbei von dem Automatisierungsgrad der Agenten abhängig.⁹⁸⁷ Der Automatisierungsgrad bezeichnet den Grad des Umfangs, in welchem der Betreiber eines Software-Agenten in die Verhandlungen eingreifen muss.⁹⁸⁸ Spezifiziert der Anwender lediglich sein Transaktionsbedürfnis, so ist der Automatisierungsgrad hoch.⁹⁸⁹ Muss der Anwender darüber hinaus noch weitere Aktionen vornehmen, um die Verhandlungen zum Abschluss zu führen, so nimmt der Automatisierungsgrad mit dem Umfang dieser Aktionen ab.

Eine zentrale Frage ist, welche Informationen den Software-Agenten bereitgestellt werden.⁹⁹⁰ Ein Ansatz besteht darin, den Akteuren grundsätzlich alle Informationen über das Marktgeschehen zur Verfügung zu stellen.⁹⁹¹ Hierbei ist zu beachten, dass auf diesem Wege zwar Informationsasymmetrien vermieden werden, es sich jedoch Probleme hinsichtlich der Vertraulichkeit, Übertragung, Speicherung und Verarbeitung der zur Verfügung gestellten Informationen ergeben können.⁹⁹² Außerdem wächst im Falle vollständiger Information der Marktteilnehmer die zu übertragende Datenmenge mit der Anzahl der Marktteilnehmer.⁹⁹³ Die Modellierung der zu übertragenden Informationen ist somit eng mit der Modellierung des Automatisierungskonzepts verbunden.⁹⁹⁴ PETERS

⁹⁸³ Vgl. PETERS (2002), S. 132.

⁹⁸⁴ Vgl. PETERS (2002), S. 131.

⁹⁸⁵ Vgl. PETERS (2002), S. 131.

⁹⁸⁶ Vgl. PETERS (2002), S. 132.

⁹⁸⁷ Vgl. PETERS (2002), S. 132.

⁹⁸⁸ Vgl. PETERS (2002), S. 133.

⁹⁸⁹ Vgl. PETERS (2002), S. 133.

⁹⁹⁰ Vgl. PETERS (2002), S. 134.

⁹⁹¹ Vgl. PETERS (2002), S. 134.

⁹⁹² Vgl. PETERS (2002), S. 134.

⁹⁹³ Vgl. PETERS (2002), S. 134.

⁹⁹⁴ Vgl. PETERS (2002), S. 134.

merkt an, dass eine separate Modellierung der beiden Aspekte nicht sinnvoll möglich ist.⁹⁹⁵

Das bei den potentiellen Anwendern nötige Vertrauen zur Nutzung autonomer Software-Agenten und der damit einhergehenden, vollständigen Delegation der Verhandlungen an diese Software-Agenten stellt eine hohe Einstiegshürde dar.⁹⁹⁶

⁹⁹⁵ Vgl. PETERS (2002), S. 134.

⁹⁹⁶ Vgl. PETERS (2002), S. 139.

6 Konzeption eines dezentralen agentenorientierten elektronischen Marktplatzes

6.1 Anforderungen

Nach Vorstellung der relevanten begrifflichen und konzeptionellen Grundlagen wird im folgenden Kapitel die Konzeption des AFEX-Systems behandelt.⁹⁹⁷ Es soll die Frage beantwortet werden, wie auf Basis der erläuterten begrifflichen und konzeptionellen Grundlagen die identifizierten Desiderata realisiert werden können. Zunächst müssen die Herausforderungen der Konzeption eines dezentralen Netzwerks beleuchtet werden.

Das Ziel des vorliegenden Kapitels ist die Konzipierung einer automatisierten Börse in Form eines elektronischen Marktplatzes. Dieser Marktplatz soll ohne die zentrale Figur eines Betreibers funktionieren und dezentral organisiert sein. Er wird als Multi-Agenten-System konzipiert und implementiert. Die Agenten müssen in der Lage sein ohne zentralen Betreiber ein funktionsfähiges, multi-laterale Kommunikation ermöglichendes Netzwerk zu bilden. Zu diesem Zweck müssen sie sich ohne zentralen Knotenpunkt („Server“) im Internet auffinden und austauschen können. Ferner sollte jeder Agent kontextabhängig sowohl als Anbieter als auch als Nachfrager auftreten können.

Durch das Handeln der Agenten entsteht ein elektronischer Marktplatz innerhalb dessen sich Angebote für und Nachfragen nach Transportdienstleistungen gegenüber stehen. Die Versteigerung von multimodalen Transportdienstleistungen mit Hilfe von zweiseitigen kombinatorischen Auktionen soll ermöglicht werden. Transportdienstleistungen werden hier als Güter gehandelt. Wie bereits beschrieben ist für jeden elektronischen Marktplatz die digitale Beschreibbarkeit der gehandelten Güter eine Voraussetzung. Ferner müssen zur Überprüfung der Kompatibilität zwischen den Handelspräferenzen einzelner Marktteilnehmer die gehandelten Güter in einem einheitlichen Format beschrieben sein. Es müssen im Rahmen der Konzeption Beschreibungssprachen definiert werden, die es ermöglichen, Transportdienstleistungen und die ihnen zugrundeliegenden Transporte in standardisierter Weise zu beschreiben. Zur Beschreibung der die Transporte bildenden Relationen (Start- und Zielorte aller Transportabschnitte) muss ein Verkehrsnetz modelliert werden, welches die Realität abstrakt abbildet. Ohne diese digitalen Repräsentationen der zu versteigernden Transportdienstleistungen und des ihnen zugrundeliegenden Verkehrsnetzes ist eine automatisierte Versteigerung derselben nicht möglich.

Die zu implementierenden Software-Agenten müssen von einem menschlichen Benutzer steuerbar sein. Zu diesem Zweck sollten sie Präferenzen besitzen, die Angebote für und Nachfragen nach Transportdienstleistungen repräsentieren. Auf Basis dieser Präferenzen

⁹⁹⁷ Die in den folgenden Kapiteln dargestellten Überlegungen und Argumentationen zur Konzeption und Implementierung des AFEX-Systems wurden in weniger detaillierter Form bereits an anderer Stelle publiziert. Vgl. FÖHRING/ZELEWSKI (2015a), FÖHRING/ZELEWSKI (2015b), FÖHRING/ZELEWSKI (2015c), FÖHRING/ZELEWSKI (2015d), sowie FÖHRING/ZELEWSKI (2015e).

können die Agenten untereinander paarweise eine Kompatibilität ihrer Präferenzen feststellen und sich auf Basis komplementärer Präferenzen zu Gruppen zusammenfinden. Innerhalb dieser Gruppen können die beschriebenen Transportdienstleistungen im Rahmen einer Auktion versteigert werden.

Zur Auktion der innerhalb einer Gruppe gehandelten Güter muss aus dem Kreis der Gruppenmitglieder ein Auktionator bestimmt werden, da im Gegensatz zu klassischen Marktplätzen kein zentraler Auktionator existiert. Der Agent, dem die Rolle des Auktionators zufällt, führt die kombinatorische Auktion durch. Nach Abschluss der Auktion stellt er einen Konsens über das Auktionsergebnis her und löst die Gruppe auf.

Die folgenden Kapitel beschreiben die soeben identifizierten Aspekte detaillierter und bilden die konzeptionelle Grundlage für die in Kapitel 7 beschriebene, prototypische Implementierung.

6.2 Multi-Agenten-Systeme als dezentrale Marktplätze

6.2.1 Herausforderungen der Dezentralisierung

Klassische Marktplätze sind zentral organisiert: Eine bestimmte Anzahl Händler trifft sich an einem festen Ort zu einer festen Zeit und begegnet dort ihren potentiellen Kunden. Im Falle einer Börse, auf der sowohl Anbieter als auch Nachfrager Transaktionswünsche formulieren und einander mitteilen können, ist es für die erfolgreiche Kontraktanbahnung entscheidend, Geschäftspartner mit möglichst kompatiblen Präferenzen zu finden, um Verhandlungen über ein mögliches Geschäft beginnen zu können.

Wie bereits beschrieben wurde haben elektronische Marktplätze dieses Schema des zentralen, physikalischen Ortes an einen virtuellen Ort verlegt und die zeitlichen Restriktionen (Öffnungszeiten) aufgehoben.⁹⁹⁸ Das Prinzip der zentralen Organisation bleibt jedoch oft erhalten.

So gibt es bei elektronischen Marktplätzen regelmäßig einen Marktplatzbetreiber, welcher die Rahmenbedingungen des stattfindenden Handels einseitig festlegt. Der Marktplatzbetreiber dient als Intermediär bei der Durchführung aller Transaktionen, er stellt die Infrastruktur des Marktplatzes, wie bspw. die Webseite, die Abwicklung finanzieller Transfers und teilweise sogar Zusatzdienstleistungen, wie bspw. ein Reklamationssystem, zur Verfügung. Für die Bereitstellung dieser Infrastruktur erhebt der Marktplatzbetreiber eine Gebühr, entweder in Form einer transportvolumen- und transaktionsabhängigen oder einer nutzer- und zeitabhängigen Gebühr.⁹⁹⁹ Während das erste Gebührenmodell auf einer Abgabe pro Transaktion oder dem beförderten Volumen basiert und somit einen höheren

⁹⁹⁸ Vgl. Kapitel 5.1.3, S. 62.

⁹⁹⁹ Für den Markt straßengebundener Güterverkehre haben BRUNS et al. diese beiden Modelle, das „transportvolumen- und transaktionsabhängige Modell“ sowie das „nutzer- und zeitabhängige Modell“, identifiziert. Vgl. BRUNS et al. (2012a), S. 7.

Ertrag erwirtschaftet, je höher die Nutzung des betreffenden Marktplatzes ausfällt, basiert das zweite Gebührenmodell auf einer reinen Zugangsbeschränkung zum Marktplatz. Die Nutzer entrichten eine Gebühr zur Nutzung unabhängig davon, wie intensiv sie das dadurch erworbene Nutzungsrecht tatsächlich ausnutzen.

Abbildung 16 zeigt die schematische Darstellung eines zentral organisierten Marktplatzes. Bei einer zentralen Marktplatzarchitektur sind alle Anbieter und Nachfrager über wechselseitige, aktive Verbindungen mit einer zentralen Plattform verbunden. Eine aktive Verbindung bedeutet, dass Daten zwischen den betreffenden Akteuren ausgetauscht werden.¹⁰⁰⁰

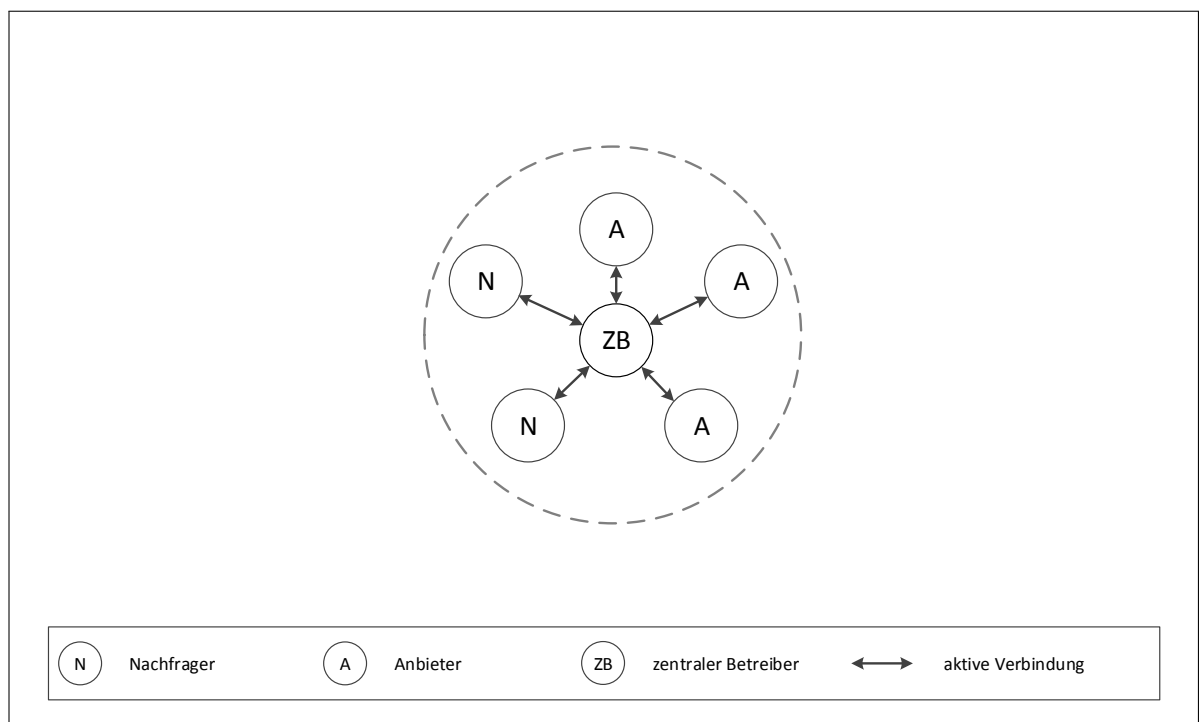


Abbildung 16: Beispiel für eine zentrale Architektur¹⁰⁰¹

Alle Interaktionen laufen über die Handelsplattform, deren Betreiber als Mittelsmann auftritt. Die Akteure agieren nicht direkt miteinander. Alle Akteure müssen alle Verhandlungen über den Marktplatz und damit den Marktplatzbetreiber führen.

Abbildung 17 zeigt im Gegensatz dazu die schematische Darstellung eines dezentral organisierten Marktplatzes.¹⁰⁰² Bei einer dezentralen Marktplatzarchitektur sind alle Anbieter

¹⁰⁰⁰ Analog werden über eine inaktive Verbindung keinerlei Daten ausgetauscht. Beide Verbindungsarten kennzeichnet, dass die betreffenden Akteure Kenntniss von der Existenz des jeweils anderen haben.

¹⁰⁰¹ Quelle: eigene Darstellung.

¹⁰⁰² Es soll in dieser Forschungsarbeit stets von einem dezentral organisierten Marktplatz (in Abgrenzung zu einem zentral organisierten) gesprochen werden, wenngleich es sich streng genommen um einen dezentral organisierten Markt handelt. Es wird jedoch davon ausgegangen, dass die beteiligten Akteure durch ihre Interdependenz und den Grad der Koordination ihrer Interaktionen einen (quasi virtuellen) Marktplatz bilden.

und Nachfrager über wechselseitige, aktive und inaktive Verbindungen miteinander verbunden. Eine aktive Verbindung repräsentiert einen aktuell stattfindenden Informationsaustausch. Eine inaktive Verbindung repräsentiert zwei Akteure, welche die Möglichkeit zur Kommunikation miteinander besitzen (bspw. durch Kenntnis einer nötigen Kontaktadresse), von dieser Möglichkeit jedoch zum aktuellen Zeitpunkt keinen Gebrauch machen.

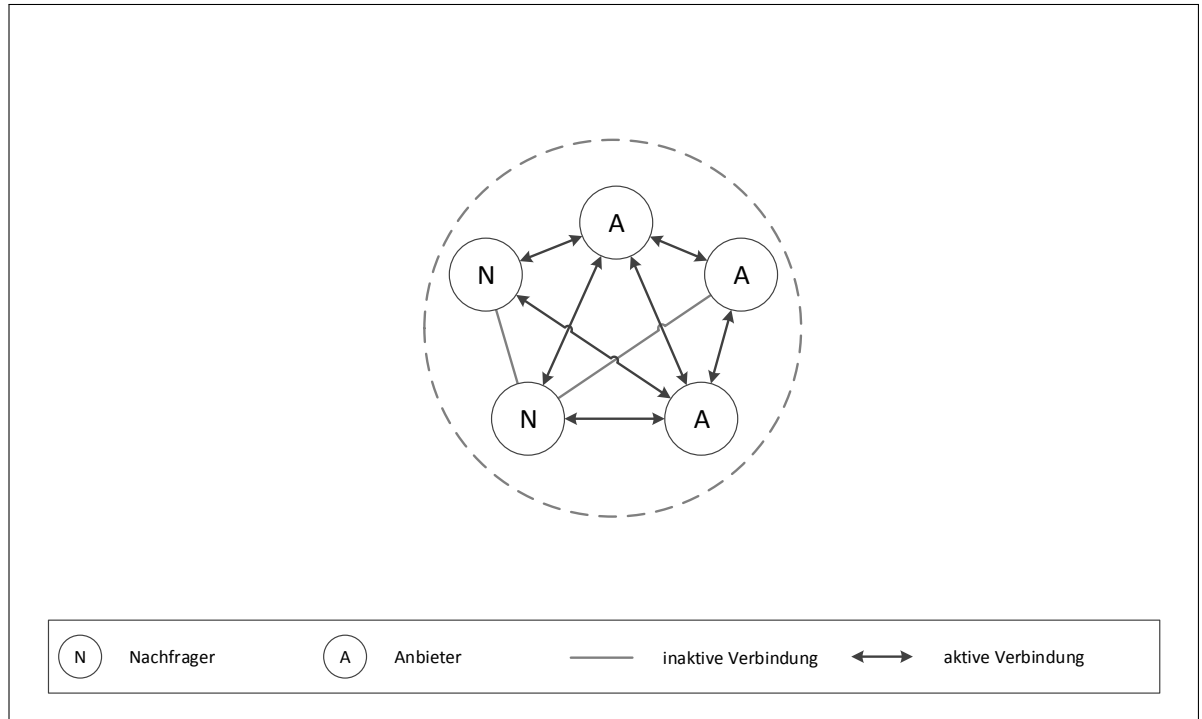


Abbildung 17: Beispiel für eine dezentrale Architektur¹⁰⁰³

Alle Interaktionen finden direkt mit den erforderlichen Akteuren statt. Im Gegensatz zu einem zentral organisierten Marktplatz gibt es keinen Mittelsmann. Alle Akteure agieren direkt miteinander und es existiert kein zentraler, virtueller Ort, an dem die Nachrichten der Akteure zusammenkommen.

Die Akteure können als Agenten im Sinne von Kapitel 5.2 gesehen werden.¹⁰⁰⁴ Sie schaffen sich eine Umgebung im Sinne der konzeptionellen Grundlagen zu Multi-Agenten-Systemen, welche für sie die beobachtbare Einflussphäre darstellt.¹⁰⁰⁵ Die Agenten müssen sich zum Zwecke des Datenaustausches koordinieren.¹⁰⁰⁶

¹⁰⁰³ Quelle: eigene Darstellung.

¹⁰⁰⁴ Vgl. Kapitel 5.2, S. 74.

¹⁰⁰⁵ Vgl. Kapitel 5.2.5, S. 85.

¹⁰⁰⁶ Vgl. Kapitel 5.2.7, S. 90.

6.2.2 Grundlagen der Kontakthanbahnung

Um ein Multi-Agenten-System zu entwickeln, welches in der Lage ist, sich ohne zentralen Knotenpunkt zu koordinieren, muss zunächst das Auffinden anderer Handelspartner ermöglicht werden. Dies ist ein nicht-triviales Problem, da eine zentrale Instanz zur Vermittlung des Kontakts zwischen den Agenten fehlt.

Dieses „Kontaktproblem“ kann jedoch gelöst werden, wenn die Agenten-Software die manuelle Eingabe von Agentenadressen ermöglicht. Eine Agentenadresse beschreibt die nötigen Informationen zur Kontaktaufnahme mit einem anderen Agenten. Da die Agenten-Software zur Kommunikation auf die Protokollfamilie „TCP/IP“ zurückgreift, besteht jede Agentenadresse aus einem Hostnamen (bspw. „demo.afex-system.org“) oder einer IP-Adresse (bspw. „127.0.0.1“)¹⁰⁰⁷ sowie einer durch einen Doppelpunkt separierten Portnummer (bspw. 1025).

Eine gültige Agentenadresse wäre bspw.

127.0.0.1:1025

Nur mit Hilfe dieser Informationen können Agenten im Internet untereinander in Kontakt treten. Sobald ein Agent sich mit Hilfe dieser Kontaktdaten mit einem anderen Agenten verbindet, tauschen sie alle ihnen bekannten Kontakte miteinander aus. Die Agenten bedienen sich zu diesem Zweck einer Kommunikationssprache.¹⁰⁰⁸

Der Vorteil dieser Vorgehensweise liegt darin, dass einzelne Agenten über nur einen Kontakt sukzessiv das gesamte, den anderen Agenten bekannte Netzwerk kennenlernen. Vor dem Beenden der Agenten-Software müssen die zur Laufzeit neu gefundenen Kontakte derart gespeichert werden, dass sie bei einem erneuten Start der Software ohne erneute Benutzereingabe kontaktiert werden können.

6.2.3 Funkfeuer-Agenten

Wesentlich benutzerfreundlicher lässt sich das beschriebene Kontaktproblem lösen, wenn die Software-Agenten sich ohne Eingreifen eines Benutzers im Internet auffinden.

Zu diesem Zweck sollte es im Internet eine oder mehrere vordefinierte Agenten-Instanzen geben, deren feste Adressen in die Agenten-Software eingebettet werden. Diese vordefinierten Agenten besitzen keine Handelspräferenzen, sondern dienen als eine Art Funkfeuer, d.h. sie leiten lediglich Kontakthanfragen weiter. Wenn eine Liste dieser „Funkfeuer-Agenten“ in alle Agenten eingebettet ist und ihnen nach der Installation zur Verfügung steht, können diese ohne Eingreifen des menschlichen Benutzers kontaktiert werden. Da

¹⁰⁰⁷ Die Beispieladresse basiert auf dem Internet Protocol Version 4 (IPv4). Es spricht aus technischer Sicht nichts gegen die Verwendung von Adressen des moderneren Internet Protocol Version 6 (IPv6). Vgl. DEERING/HINDEN (1998).

¹⁰⁰⁸ Vgl. Kapitel 5.2.3, S. 79.

die Agenten einmal bekannte Kontaktinformationen speichern, sind sie in der Lage, die über den Funkfeuer-Agenten gefundenen Kontakte zu einem späteren Zeitpunkt erneut anzusprechen, ohne auf den oder die Funkfeuer-Agenten als Kontaktvermittler angewiesen zu sein.

Das Beispiel in Abbildung 18 verdeutlicht den Prozess:

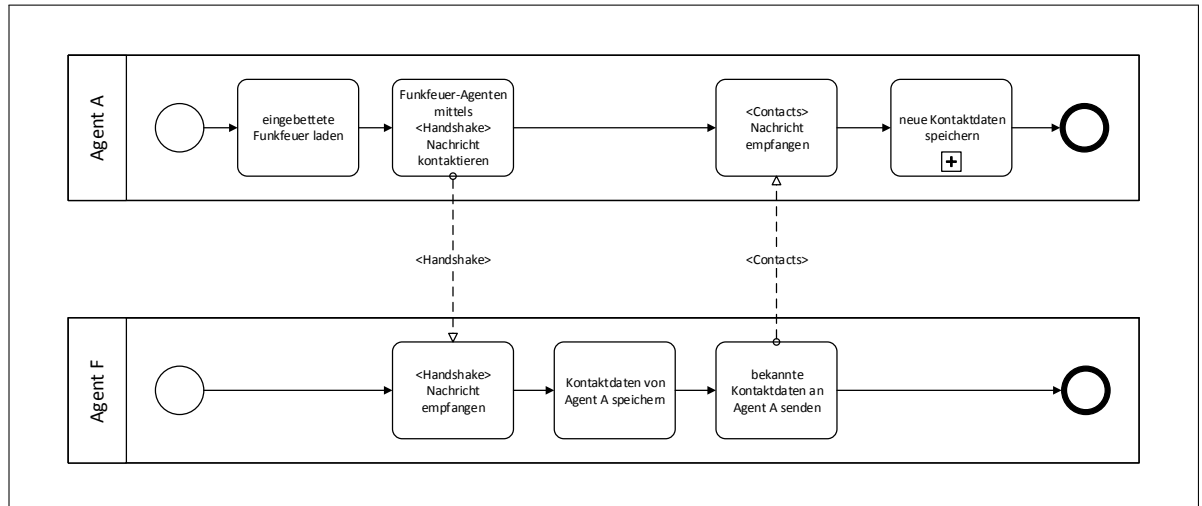


Abbildung 18: Kontaktaufnahme unter Agenten mit Hilfe eines Funkfeuer-Agenten¹⁰⁰⁹

- Agent *A* kennt noch keine anderen Agenten außer den Funkfeuer-Agenten *F*.
- Agent *A* kontaktiert *F* und erhält von ihm weitere Agentenadressen, bspw. jene von Agent *B* und Agent *C*.
- Agent *A* speichert diese Kontakte.
- Auch wenn das Programm des Agenten *A* beendet wird, kann dieser Agent die gespeicherten Kontakte beim nächsten Start erneut abrufen.

Der Vorteil dieser Vorgehensweise ist, dass Funkfeuer-Agenten unabhängig voneinander betrieben, kommuniziert und genutzt werden können.¹⁰¹⁰ Sie unterstützen die dezentrale Organisation des AFEX-Systems, da sie einerseits das beschriebene Kontaktproblem lösen ohne eine Benutzerinteraktion zu erfordern, und andererseits für den Betrieb des dezentralen Marktplatzes nicht notwendig sind: Die menschlichen Benutzer könnten stets über die manuelle Eingabe ein „eigenes“ Kontaktnetzwerk aufbauen.

¹⁰⁰⁹ Quelle: eigene Darstellung.

¹⁰¹⁰ Eine Analogie in der realen Welt sind verschiedene per Telefon ansprechbare Auskunftsdienste, welche auf Anfrage die Telefonnummer eines Teilnehmers zurückgeben. Die zurückgegebene Nummer ist hier im Regelfall, unabhängig von der gewählten Auskunft, stets dieselbe und das Telefonnetz funktioniert auch ohne Auskunftsdienste, solange einer von zwei Teilnehmern die Telefonnummer des anderen kennt.

6.2.4 Rundruf-Kanäle

Eine öffentlichere Möglichkeit zur asynchronen Kontaktvermittlung sind „Rundruf-Kanäle“. Rundruf-Kanäle sind öffentlich zugängliche und frei verfügbare Kommunikationskanäle, in denen die Agenten eine auf Kontakthanbahnung abzielende, speziell formatierte Nachricht absetzen können. Sobald andere Agenten diese Nachricht über den betreffenden Rundruf-Kanal empfangen, werden sie dazu veranlasst, den sendenden Agenten außerhalb des Rundruf-Kanals über die in der Nachricht enthaltenen Informationen zu kontaktieren.

Hierbei nutzen die Agenten öffentlich zugängliche Chaträume, Foren, Messageboards oder *social media* Kanäle, um sich miteinander bekannt zu machen: Die Agenten registrieren sich bei einem öffentlich zugänglichen Dienst, wie bspw. freien IRC-Servern, bulletin boards und Twitter und verbreiten eine öffentlich auffindbare Nachricht, die mehrere Schlüsselwörter enthält, anhand derer andere Agenten Kontakt zu ihnen aufnehmen können.

Das Beispiel in Abbildung 19 verdeutlicht den Prozess:

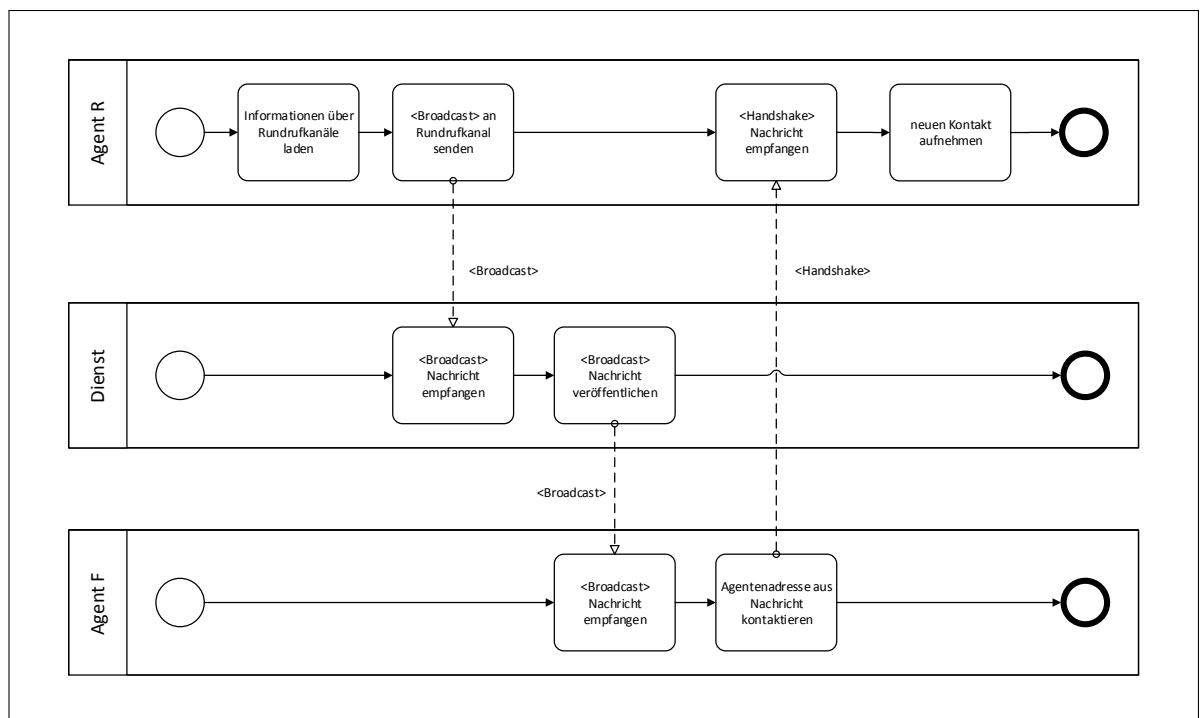


Abbildung 19: Kontaktaufnahme unter Agenten mit Hilfe eines Rundruf-Kanals¹⁰¹¹

Agent *R* meldet sich bei einem allen Agenten bekannten, frei zugänglichen Dienst, bspw. einem IRC-Server, an. Er eröffnet einen Chatraum mit einem zuvor in der Software hinterlegten Namen (bspw. „AFEX-Broadcast“). Existiert der Chatraum bereits, so tritt *R* ihm bei. Hier veröffentlicht er eine Nachricht nach einem zuvor vereinbarten Schema.

¹⁰¹¹ Quelle: eigene Darstellung.

Eine solche Nachricht könnte mit einer den Agenten bekannten Zeichenkette beginnen (bspw. „contact:“) und enden (bspw. „#afexbroadcast“). Zwischen diesen Schlüsselwörtern könnte der Agent seine Agentenadresse weitergeben.

Eine nach diesem Schema aufgebaute Nachricht wäre bspw.

contact: 127.0.0.1:1025 #afexbroadcast

Agent F , der sich bereits im Chat aufhält, sieht diese Nachricht und kontaktiert R über die in der Nachricht enthaltenen Informationen außerhalb des Chatraums. R und F verbleiben in dem Chatraum, um die Nachrichten von Agenten lesen zu können, die nach ihnen beitreten.

6.2.5 Relevanz für die prototypische Implementierung

Die dargestellten Ansätze der Kontakthanbahnung bedeuten eine partielle Re-Zentralisierung für das dezentrale Netzwerk:

1. Funkfeuer-Agenten emulieren die Existenz gleich mehrerer zentraler Knotenpunkte zur Kontakthanbahnung.
2. Rundruf-Kanäle stellen einen Ansatz dar, im dezentralisierten Netzwerk verlässlich verfügbare Kommunikationskanäle zur Verfügung zu stellen.

Aus konzeptioneller Sicht stellen beide Konzepte „Ressourcen“ innerhalb der von den Agenten beobachtbaren Umgebung dar, welche einen für das Multi-Agenten-System entscheidenden Zustand (die Existenz weiterer Agenten sowie ihre Kontaktadressen) kommunizieren.¹⁰¹² Den Agenten müssen zur Nutzung dieser Ressourcen bereits zur Entwicklungszeit entsprechende Pläne implementiert werden.

6.3 Verkehrsnetz

6.3.1 Konzeptionelle Grundlagen

Zur Konzeption und Implementierung eines Systems zur Versteigerung von Transportdienstleistungen ist es nötig, das Modell eines Verkehrsnetzes zu konzipieren, innerhalb dessen die zu versteigernden Transporte abgebildet werden. Dieses Verkehrsnetz¹⁰¹³ muss in der Lage sein, die Realität so abzubilden, dass eine Versteigerung über verschiedene Teilabschnitte der Transportstrecke und eine Nutzung verschiedener Transportmittel zur Abwicklung des Transports möglich sind.

¹⁰¹² Vgl. Kapitel 5.2.5, S. 85.

¹⁰¹³ Im Folgenden werden die Begriffe „Modell des Verkehrsnetzes“ und „Verkehrsnetz“ zur leichteren Lesbarkeit synonym verwendet. Sollte mit „Verkehrsnetz“ nicht die dieser Forschungsarbeit zugrunde liegende Modellierung, sondern das real existierende Verkehrsnetz gemeint sein, so wird dies explizit benannt.

Es wird angenommen, dass Anbieter in der Lage sind, ihre Transporte in einem Verkehrsnetz detailliert zu beschreiben, die meisten Nachfrager ihre Bedürfnisse jedoch nur auf einem sehr hohen Abstraktionsgrad beschreiben können bzw. wollen. Dies begründet sich in der Plausibilitätsüberlegung, dass ein Anbieter eine überschaubare Anzahl möglicher Verbindungen an den von ihm angefahrenen Bahnhöfen, Lieferadressen etc. anbietet. Es wird vermutet, dass Nachfrager im Gegensatz dazu regelmäßig Bedürfnisse haben, deren Erfüllung durch konkurrierende Angebote mehrerer Anbieter möglich ist oder deren Erfüllung sogar die Kooperation mehrerer Anbieter erforderlich macht. Im Gegensatz zu den Anbietern, welche ihre Transportdienstleistungen auf genau die im System hinterlegte Art und Weise zu erfüllen im Stande sind, werden Nachfrager regelmäßig indifferent sein, durch welchen und wie viele Anbieter die Erfüllung ihrer Nachfrage erfolgt. Somit müssen zur formalen Beschreibung einer Nachfrage alle alternativ in Betracht kommenden Angebote berücksichtigt und alle Kombinationen möglicher Teilstrecken gefunden werden. Die Formulierung einer derartigen Nachfrage ist auf formaler Ebene komplexer als die Formulierung der meisten Angebote.

Daher erscheint es sinnvoll, dass die Anbieter alle ihre Angebote – auf das im Folgenden modellierte Verkehrsnetz abgestimmt – im System hinterlegen. Das System hilft dabei, die formalsprachlich komplexe Anfrage eines Nachfragers in die Form der Angebotsdefinition zu überführen und mit den vorhandenen Angeboten abzugleichen.

Praktisch bedeutet dies, dass alle Anbieter ihre Angebote in einer vorgegebenen Notation im System hinterlegen müssen. Die Nachfrager müssen nur angeben, dass sie eine Transportdienstleistung von Haltepunkt H_1 nach H_2 suchen.

Der folgenden Modellierung liegen somit folgende Prämissen zugrunde:

1. Unter den Teilnehmern existieren mehr Nachfrager als Anbieter.¹⁰¹⁴
2. Die von Teilnehmern formulierten Nachfragen nach Transportdienstleistungen sind formalsprachlich ausgedrückt regelmäßig komplexer als die von Teilnehmern formulierten Angebote.
3. Das Verkehrsnetz berücksichtigt verschiedene Verkehrsträger.
4. Alle Transporte lassen sich in Transportabschnitte unterteilen, welche zwischen jeweils zwei Haltepunkten stattfinden, die durch einen Verkehrsweg verbunden sind.

Zur Berücksichtigung dieser Prämissen wird das gesamte reale Verkehrsnetz im ersten Schritt der Modellierung auf zwei grundlegende Elemente vereinfacht: Haltepunkte und Relationen.

¹⁰¹⁴ Vgl. HABIB/BRUNS (2012).

6.3.2 Haltepunkte

Haltepunkte sind im Sinne dieser Forschungsarbeit definiert als physische Orte des Güterumschlags, von denen mindestens ein Verkehrsweg zu mindestens einem anderen Haltepunkt führt. Typische Beispiele für Haltepunkte sind Umschlagterminals, Güterbahnhöfe und Seehäfen.

Ein *Verkehrsweg* ist ein Weg, welcher durch Benutzung eines Verkehrsträgers eine Ortsveränderung von Gütern ermöglicht. Typische Beispiele für Verkehrswege sind Straßen, Schienenwege und Wasserstraßen.

Verkehrsträger unterteilen sich in solche mit überschaubar vielen, festen Haltepunkten und solche mit unüberschaubar vielen, flexiblen Haltepunkten.

Typische Beispiele für die erste Art von Verkehrsträgern sind Schienen-, Schiffs- und Flugverkehr. Das typische Beispiel für die zweite Art von Verkehrsträgern ist der Straßenverkehr und hierbei primär der Transport per Lkw.¹⁰¹⁵

Typische Beispiele für die erste Art von Haltepunkten sind Güterbahnhöfe und Seehäfen. Ein typisches Beispiel für die zweite Art von Haltepunkten sind postalische Lieferadressen.

Folgende Notation wird zur Erfassung dieser Haltepunkte verwendet:

1. H_i für feste Haltepunkte. Sie werden durch Großbuchstaben dargestellt, die einen Index i zur Identifikation des einzelnen Haltepunkts tragen.
2. H_i^r für eine Klasse flexibler Haltepunkte. Sie werden als „Haltepunkte innerhalb eines festen Radius r um einen festen Haltepunkt H_i “ definiert. Sie werden in der Notation des referenzierten, festen Haltepunktes dargestellt und um die Angabe der Entfernung r erweitert. Individuelle Haltepunkte innerhalb der Klasse H_i^r werden durch die Ergänzung eines Index a kenntlich gemacht: ${}_aH_i^r$

Mit Hilfe dieser Vereinbarungen lässt sich bspw. ein Gleisanschluss, Hafen oder Flugplatz als H_1 darstellen. Der Haltepunkt eines Warenlagers, welches 7 km von H_1 entfernt liegt und nur per Lkw zu erreichen ist, lässt sich z.B. durch ${}_1H_1^7$ darstellen.

Selbstverständlich eignet sich diese Notation fester und flexibler Haltepunkte nicht, um einen Ort, wie bspw. das hypothetische Warenlager in Hafennähe, exakt zu verorten, da die Angabe ${}_aH_i^r$ lediglich aussagt, dass sich ein Haltepunkt im Umkreis von höchstens r Kilometern um H_i befindet. Die Notation bietet vielmehr die Möglichkeit, Verbindungen zwischen zwei Haltepunkten zu beschreiben, da sie den Anbietern von Transporten die

¹⁰¹⁵ Es kommen für die zweite Art von Verkehrsträgern prinzipiell auch exotischere Verkehrsmittel infrage, wie bspw. spezielle Lufttransporte per Lasthubschrauber oder Kleinsttransporte per Fahrradkurrier. Diese Verkehrsträger sind durch die getroffene Definition zwar gedeckt, werden jedoch aufgrund ihrer verhältnismäßig geringen Bedeutung im Vergleich zum Lkw-basierten Straßengüterverkehr nicht weiter berücksichtigt.

Flexibilität bietet, sowohl feste Strecken als auch flexible Transporte im Umkreis um einen festen Punkt zu beschreiben.

Abbildung 20 verdeutlicht dies. Sie zeigt einen festen Haltepunkt und einige flexible Haltepunkte. H_7 stellt einen festen Haltepunkt dar und befindet sich an einem festen Ort. H_7^2 stellt die Klasse aller flexiblen Haltepunkte ${}_aH_7^2$ im Umkreis von höchstens 2 km um den festen Haltepunkt H_7 dar.

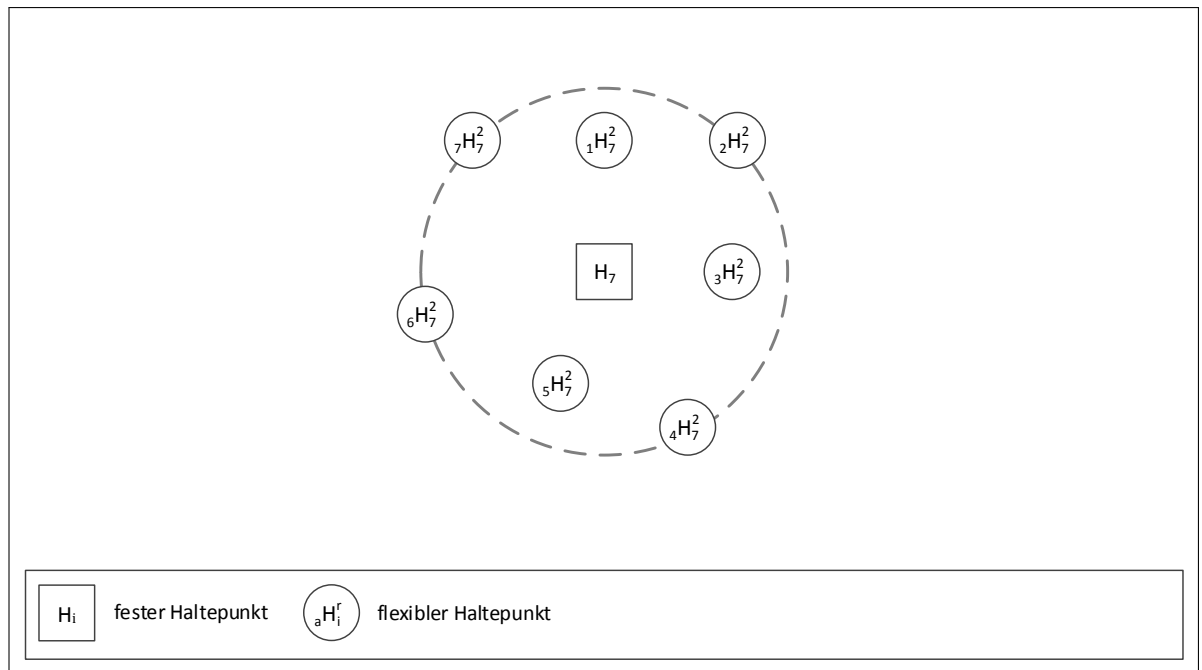


Abbildung 20: Feste und flexible Haltepunkte¹⁰¹⁶

6.3.3 Verbindungen und Relationen

Um innerhalb eines Verkehrsnetzes Verkehre zwischen Haltepunkten beschreiben zu können, ist es nötig, den Begriff der Verbindung sowie die Notation von Verbindungen zu definieren.

Eine *Verbindung* ist die Gesamtheit aller Relationen, welche einen Verkehr zwischen zwei Haltepunkten ermöglicht.

Eine *Relation* ist der direkte Verkehrsweg zwischen zwei Haltepunkten.

Aus den vorrangegangenen Überlegungen ergeben sich vier verschiedene Arten von Relationen:

1. Fester Haltepunkt zu feste Haltepunkt,
2. flexibler Haltepunkt zu fester Haltepunkt,
3. fester Haltepunkt zu flexibler Haltepunkt sowie

¹⁰¹⁶ Quelle: eigene Darstellung.

4. flexibler Haltepunkt zu flexibler Haltepunkt.

Relationen können als Tupel aus zwei Haltepunkten notiert werden:

1. (H_1, H_2) – fester Haltepunkt H_1 zu festem Haltepunkt H_2 ,
2. $({}_1H_1^r, H_1)$ – flexibler Haltepunkt im Umkreis von höchstens r Kilometern um H_1 zu festem Haltepunkt H_1 ,
3. $(H_1, {}_1H_1^r)$ – fester Haltepunkt H_1 zu flexiblem Haltepunkt im Umkreis von höchstens r Kilometern um H_1 ,
4. $({}_1H_1^r, {}_1H_2^s)$ – flexibler Haltepunkt im Umkreis von höchstens r Kilometern um H_1 zu flexiblem Haltepunkt im Umkreis von höchstens s Kilometern um H_2 .

Relationen können die Grundlage für eine Transportdienstleistung bilden, da sie Anfangs- und Endpunkt eines möglichen Transportes darstellen.

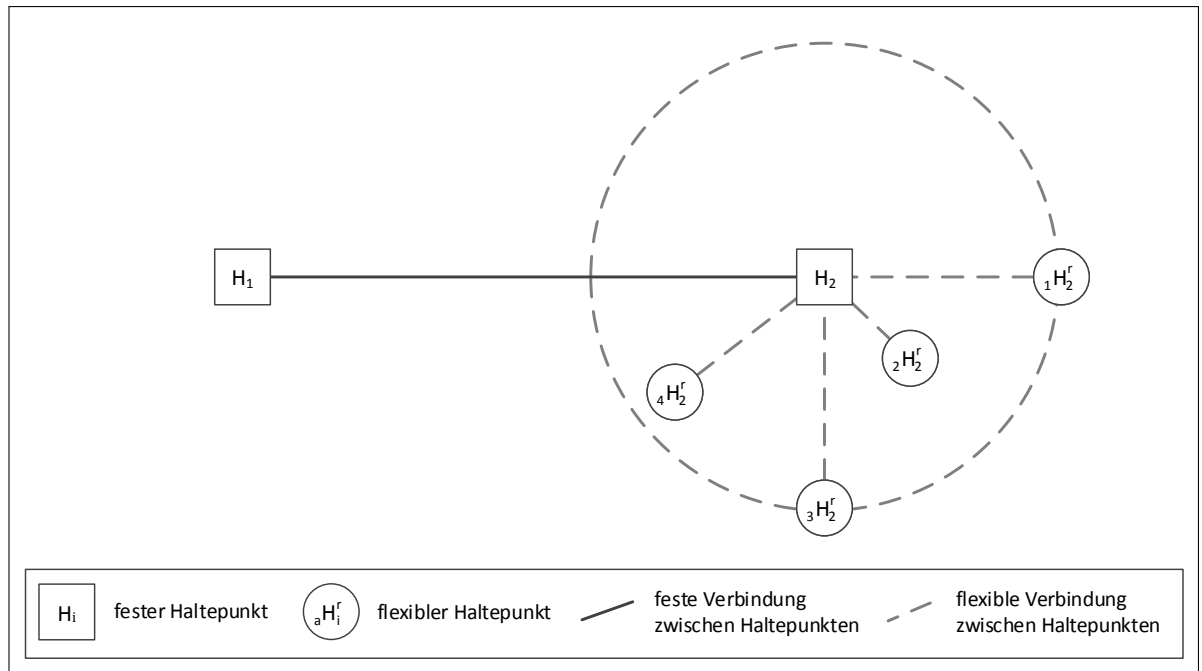
Jede Verbindung kann als Liste aus Relationen dargestellt werden, wie z.B.

$$\{(H_1, H_2); (H_2, {}_1H_2^5)\}$$

Eine feste Verbindung ist eine Verbindung, welche nur aus Relationen zwischen festen Haltepunkten besteht.

Eine flexible Verbindung besteht, wenn eine Verbindung aus Relationen besteht, von denen mindestens eine zumindest einen flexiblen Haltepunkt beinhaltet.

Abbildung 21 zeigt eine feste Verbindung zwischen den festen Haltepunkten H_1 und H_2 sowie mehrere flexible Verbindungen zwischen dem festen Haltepunkt H_2 und den flexiblen Haltepunkten ${}_aH_2^r$ mit $a = 1, \dots, 4$.

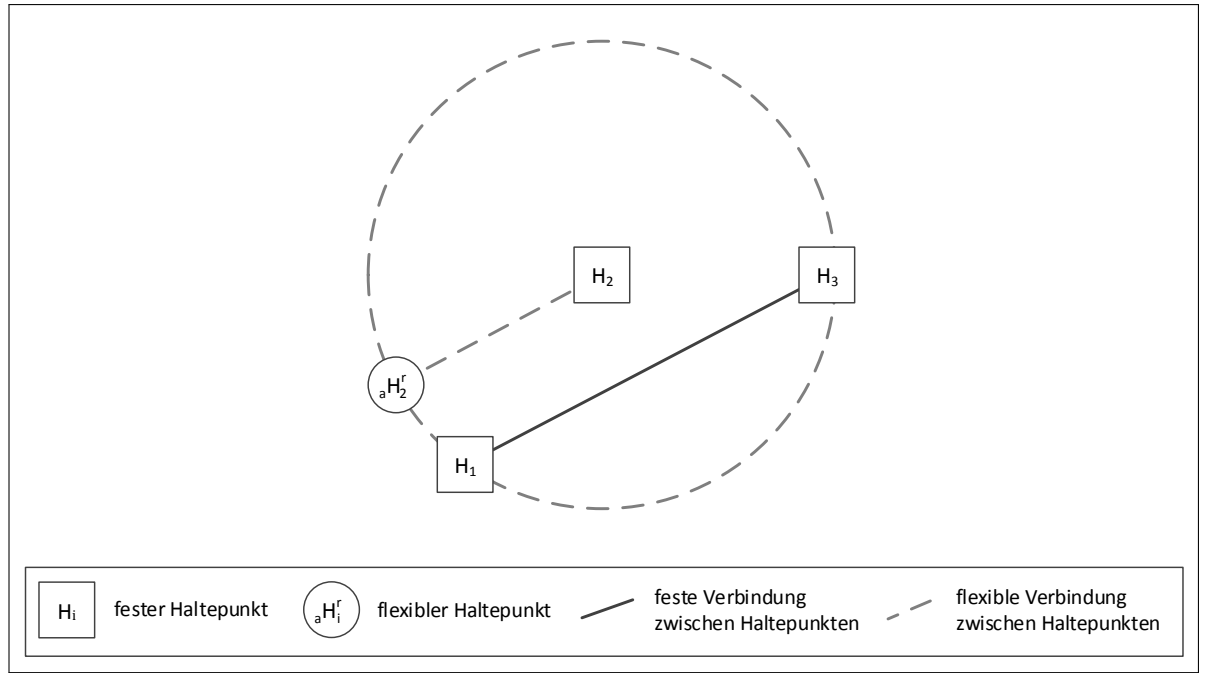
Abbildung 21: Flexible Haltepunkte¹⁰¹⁷

Die denkbaren Konstellationen, welche durch feste und flexible Verbindungen den Gütertransport zwischen zwei Haltepunkten ermöglichen, ergeben eine Reihe an Spezialfällen, die im Folgenden beschrieben werden.

Spezialfall 1

Als Spezialfall ist eine Verbindung zwischen zwei festen Haltepunkten denkbar, welche zudem durch eine Verbindung zwischen zwei flexiblen Haltepunkten abgebildet werden kann. Abbildung 22 zeigt diesen Sachverhalt: Ein Transport zwischen den festen Haltepunkten H_1 und H_3 kann durch eine Verbindung (H_1, H_3) realisiert werden. Es ist jedoch auch möglich, denselben Transport mit der Verbindung $({}_aH_2^r, {}_bH_2^r)$ zu realisieren, da beide feste Haltepunkte in einem Radius r um den Haltepunkt H_2 verortet sind. Bietet also ein Anbieter die flexiblen Transporte um H_2 an, so könnten diese eine Alternative zur festen Verbindung (H_1, H_3) sein.

¹⁰¹⁷ Quelle: eigene Darstellung.

Abbildung 22: Haltepunkte – Spezialfall 1¹⁰¹⁸

Spezialfall 2

Ein flexibler Haltepunkt kann mehrere Bezeichnungen haben, wenn sich die Radien verschiedener Angebote überschneiden. Es bieten bspw. zwei verschiedene Anbieter flexible Transporte um Haltepunkt H_1 an: Anbieter 1 bietet ${}_aH_1^7$ an, Anbieter 2 jedoch nur ${}_aH_1^5$. Nun gilt ${}_aH_1^5 \in {}_aH_1^7 \forall j \in \mathbb{N}$, da alle Punkte in ${}_aH_1^5$ auch innerhalb von ${}_aH_1^7$ liegen.

Wenn sich die Radien der flexiblen Verbindungen benachbarter fester Haltepunkte überschneiden, kann ein ähnlicher Effekt für alle in der Schnittmenge der Kreise verorteten flexiblen Haltepunkte festgestellt werden.

Abbildung 23 veranschaulicht diesen Fall: Angenommen, der Punkt ${}_aH_i^r$ liegt 5 Kilometer von H_2 und 7 Kilometer von H_3 entfernt, so ergeben sich für ihn u.a. die Bezeichnungen ${}_1H_2^5$ sowie ${}_1H_3^7$. Ein Transport von H_1 nach ${}_aH_i^r$ könnte demnach folgendermaßen realisiert werden:

$$\{(H_1, H_2); (H_2, {}_1H_2^5)\}$$

$$\{(H_1, H_3); (H_3, {}_1H_3^7)\}$$

$$\{(H_1, H_2); (H_2, H_3); (H_3, {}_1H_3^7)\}$$

¹⁰¹⁸ Quelle: eigene Darstellung.

$$\{(H_1, H_3); (H_3, H_2); (H_2, {}_1H_2^5)\}$$

Der flexible Haltepunkt ${}_aH_i^r$ trägt also in Abhängigkeit davon, welcher Transport betrachtet wird, die Bezeichnung ${}_aH_2^5$ oder ${}_aH_3^7$.

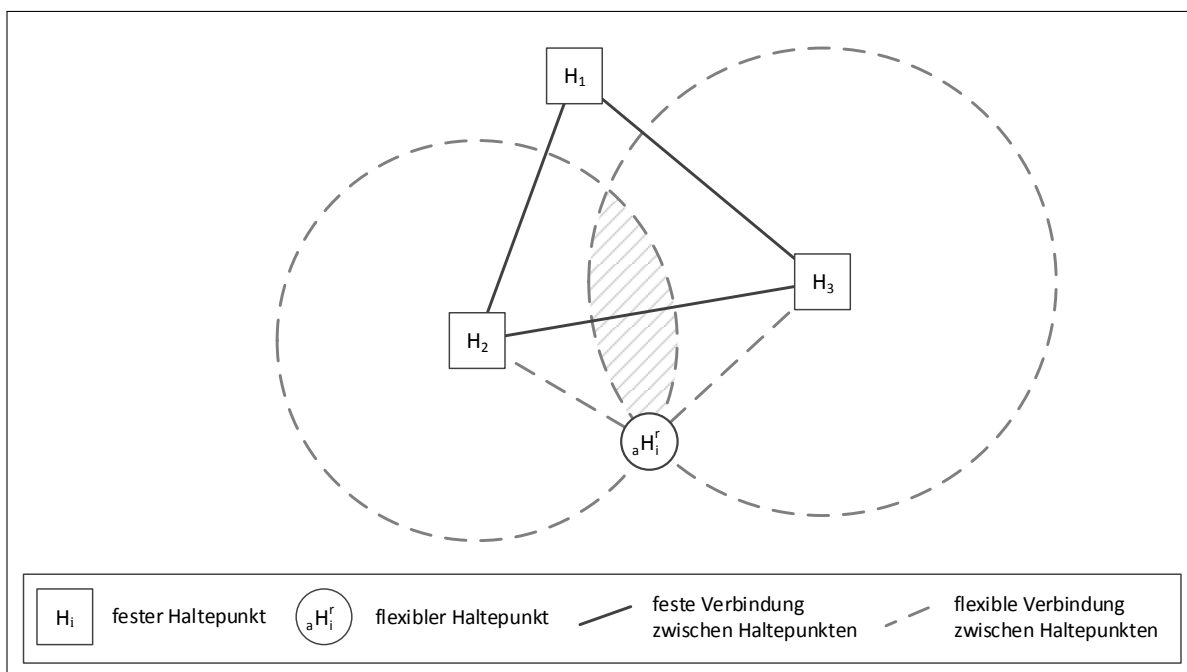
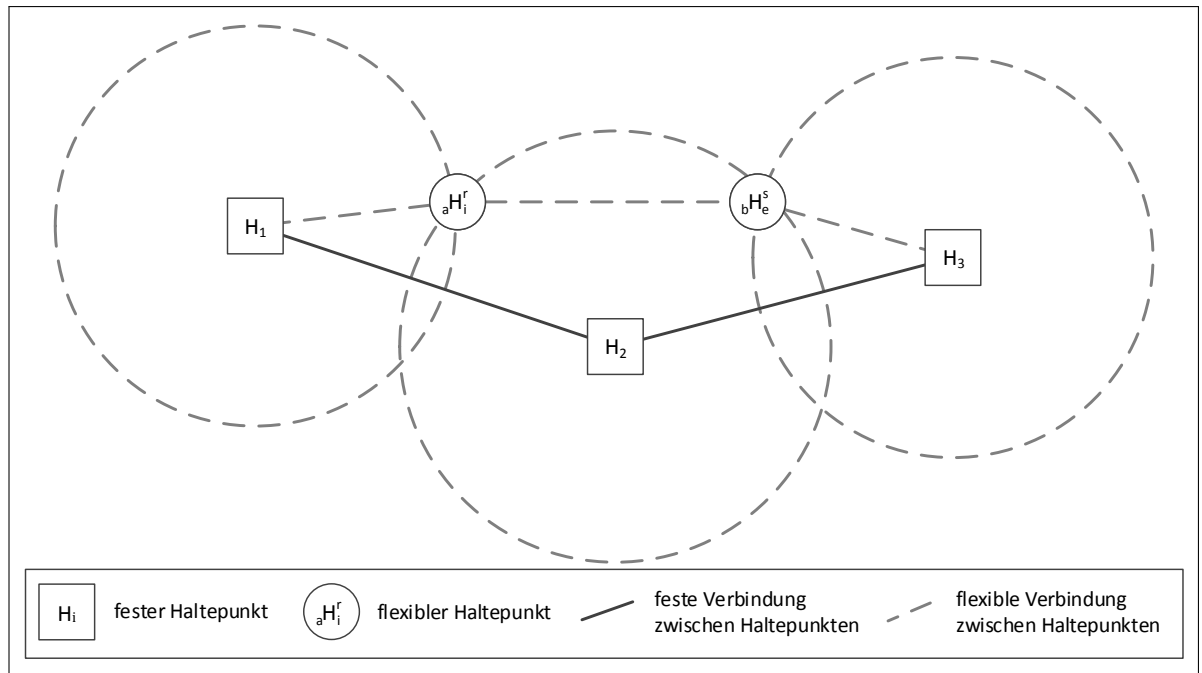


Abbildung 23: Haltepunkte – Spezialfall 2¹⁰¹⁹

Spezialfall 3

Eine Abwandlung des vorangegangenen Falls ergibt einen Fall, in welchem sogar eine aus mehreren Relationen bestehende, feste Strecke durch flexible Haltepunkte substituiert werden kann, wie Abbildung 24 zeigt.

¹⁰¹⁹ Quelle: eigene Darstellung.

Abbildung 24: Haltepunkte – Spezialfall 3¹⁰²⁰

Die Verbindung

$$\{(H_1, H_2); (H_2, H_3)\}$$

lässt sich bei Verfügbarkeit entsprechender Angebote auch über

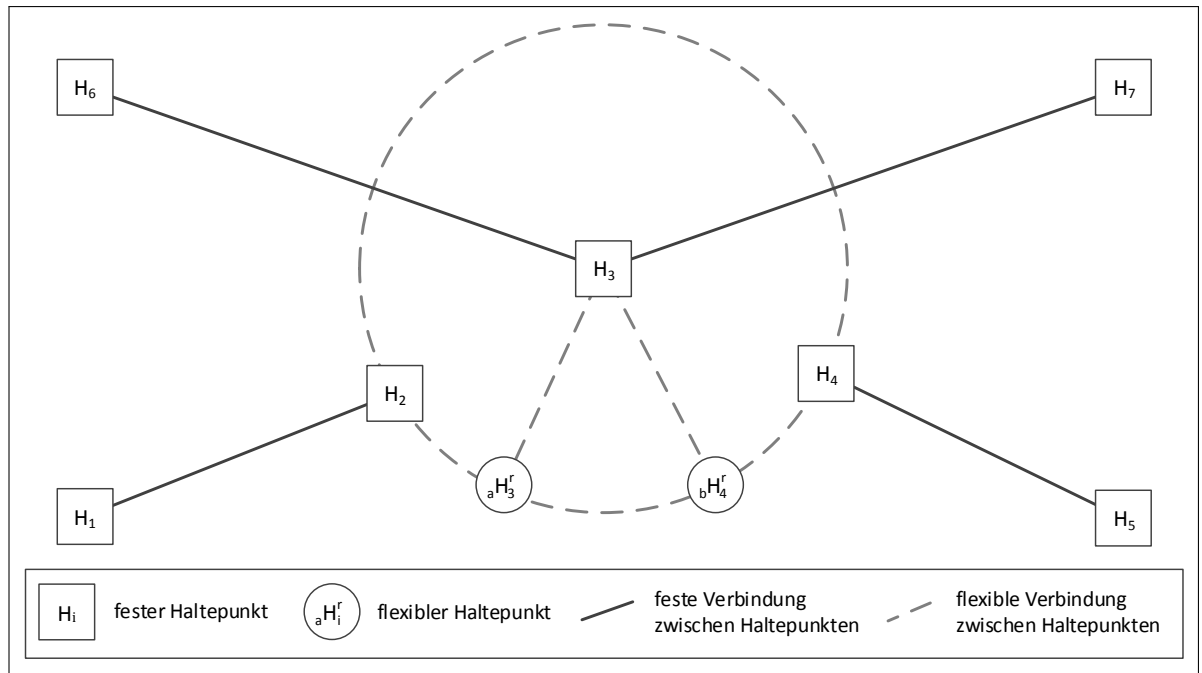
$$\{(H_1, {}_aH_i^r); ({}_aH_i^r, {}_bH_e^s); ({}_bH_e^s, H_3)\}$$

realisieren. Da jedoch in der Realität die beteiligten Speditionen in einer Weise kooperieren müssten, dass an den Haltepunkten ${}_aH_i^r$ sowie ${}_bH_e^s$ jeweils sämtliche zu transportierenden Güter umgeladen werden können, erscheint diese Art der Substitution von festen durch flexible Verbindungen eher theoretischer Natur zu sein.

Spezialfall 4

Die Kombination der zuvor beschriebenen Fälle ergibt einen Fall, in welchem eine „fehlende“ feste Verbindung durch flexible Haltepunkte geschaffen wird. Abbildung 25 zeigt diesen Sachverhalt.

¹⁰²⁰ Quelle: eigene Darstellung.

Abbildung 25: Haltepunkte – Spezialfall 4¹⁰²¹

Ein Transport von H_1 nach H_5 könnte nur dann zustande kommen, wenn die Relation $({}_aH_3^r, {}_bH_4^r)$ angeboten wird. Die Relation würde dann

$$\{(H_1, {}_aH_3^r); ({}_aH_3^r, {}_bH_4^r); ({}_bH_4^r, H_5)\}$$

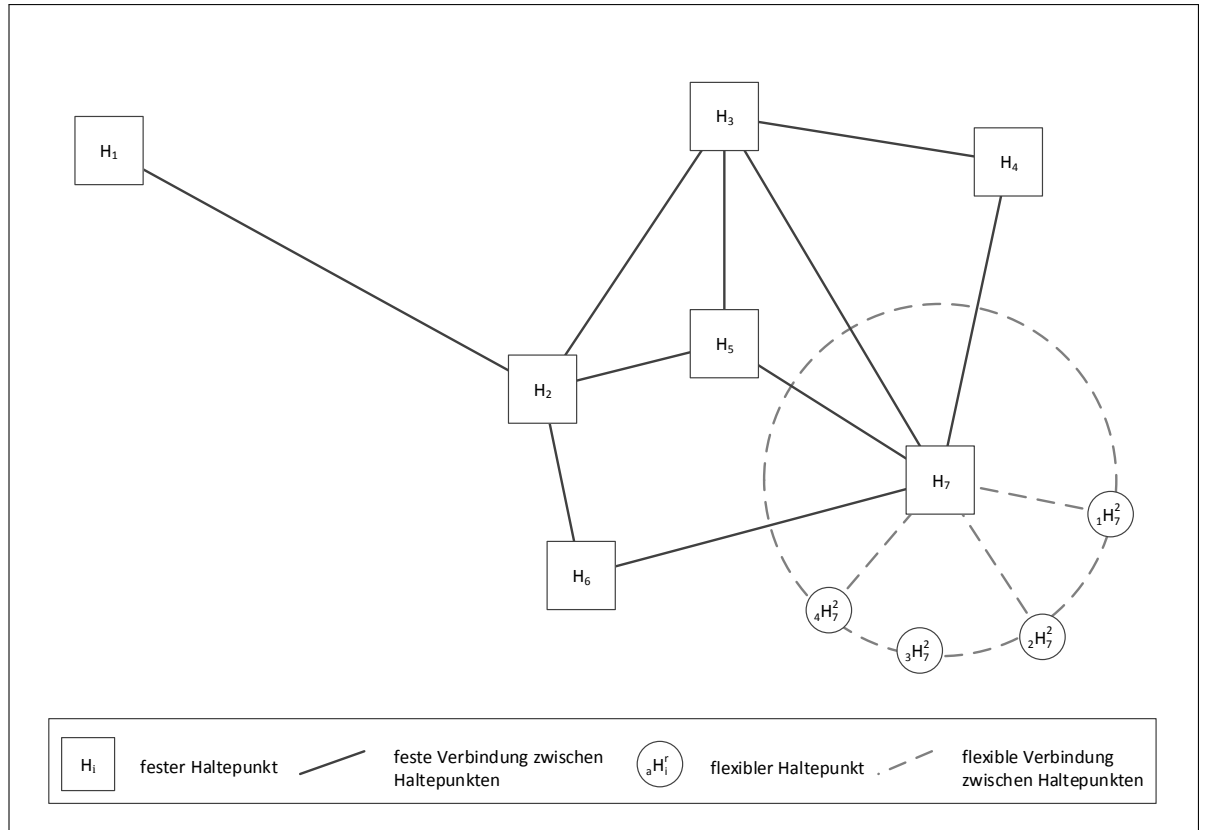
lauten, da H_2 und ${}_aH_3^r$ bzw. H_4 und ${}_bH_4^r$ den jeweils gleichen Haltepunkt bezeichnen können.

6.3.4 Anwendungsbeispiel

Abbildung 26 zeigt ein exemplarisches Verkehrsnetz unter Verwendung fester und flexibler Haltepunkte. H_1 und H_2 können hierbei Abstraktionen zweier Häfen sein, die zwei entfernte Orte miteinander verbinden. Von H_2 aus gehen mehrere feste, bspw. schienengebundene, Relationen in das lokale Verkehrsnetz.¹⁰²² H_7 zeigt schließlich einen festen Haltepunkt einschließlich flexibler Haltepunkte im Umkreis von höchstens 2 km.

¹⁰²¹ Quelle: eigene Darstellung.

¹⁰²² Gemeint sind alle Haltepunkte in Abbildung 26 ohne H_1 .

Abbildung 26: Exemplarisches Verkehrsnetz¹⁰²³

Im Anwendungsbeispiel sind drei Anbieter mit folgenden, nicht näher spezifizierten Angeboten für Transportdienstleistungen gegeben:

1. Anbieter A bietet freie Kapazitäten in einem Zug von H_2 nach H_7 an, welcher über den Knoten H_3 fährt.
2. Anbieter B bietet freie Kapazitäten in einem Schiff von H_1 nach H_2 an.
3. Anbieter C bietet Transporte im Umkreis von 2 Kilometern um H_7 an.

Zusätzlich existiert ein Nachfrager D , welcher freie Kapazitäten für den Transport eines Gutes von H_1 nach ${}_1H_7^2$ sucht.

Das System muss nun alle möglichen Relationen im gegebenen Verkehrsnetz ermitteln, die D 's Nachfrage erfüllen würden. Für das in Abbildung 26 gezeigte Verkehrsnetz sind die möglichen Verbindungen:¹⁰²⁴

$$\{(H_1, H_2), (H_2, H_6), (H_6, H_7), (H_7, {}_1H_7^2)\}$$

$$\{(H_1, H_2), (H_2, H_5), (H_5, H_7), (H_7, {}_1H_7^2)\}$$

¹⁰²³ Quelle: eigene Darstellung.

¹⁰²⁴ Aus Gründen der Vereinfachung wird davon ausgegangen, dass kein Haltepunkt mehrmals angefahren wird.

$$\{(H_1, H_2), (H_2, H_3), (H_3, H_7), (H_7, {}_1H_7^2)\}$$

$$\{(H_1, H_2), (H_2, H_3), (H_3, H_5), (H_5, H_7), (H_7, {}_1H_7^2)\}$$

$$\{(H_1, H_2), (H_2, H_3), (H_3, H_4), (H_4, H_7), (H_7, {}_1H_7^2)\}$$

Anschließend könnte das System determinieren, dass die Kombination der Angebote der Anbieter A , B und C eine mögliche Kombination zur Erfüllung der Nachfrage von Nachfrager D darstellen:

$$\underbrace{\{(H_1, H_2)\}}_B, \underbrace{\{(H_2, H_3), (H_3, H_7)\}}_A, \underbrace{\{(H_7, {}_1H_7^2)\}}_C$$

Es ist also in diesem stark vereinfachten Beispiel möglich, die von Nachfrager D nachgefragte Verbindung als Kombination aus den angebotenen Relationen darzustellen.

6.3.5 Relevanz für die prototypische Implementierung

Das beschriebene Verkehrsnetz ist für die Auktion von Transportdienstleistungen relevant, da die Verkehrswege, auf denen die Transportdienstleistungen stattfinden, standardisiert beschreibbar sein müssen, um in einem elektronischen Marktplatz gehandelt werden zu können. Die hier beschriebenen Relationen sind als Transportabschnitte zur Beschreibung von Teildienstleistungen relevant.

6.4 Agentenpräferenzen

6.4.1 Begriff und Bedeutung der Präferenzen

Damit die Agenten die gesamte nachgefragte oder angebotene Transportdienstleistung ersteigern bzw. versteigern können, müssen sie interne Präferenzen bzgl. der Transportstrecke, Fracht und Zahlungsbereitschaft besitzen.

Die Agenten-Software muss durch einen menschlichen Nutzer steuerbar sein.¹⁰²⁵ Die menschlichen Benutzer müssen in der Lage sein über eine Benutzerschnittstelle Präferenzen in Form von Geboten zu definieren, welche der Agent anschließend im Rahmen von Auktionen im Auftrag seines Benutzers abgibt.

¹⁰²⁵ Dies stellt keine neue Anforderung dar, sondern wurde bereits während der Anforderungsanalyse für den ORFE-Softwareprototyp festgelegt. Vgl. Kapitel 6.1, S. 115.

Die Präferenzen ihrer Benutzer stellen für die Agenten eine Information aus ihrer Umgebung dar, welche sie über die Benutzerschnittstelle beobachten können.¹⁰²⁶ Die Benutzerschnittstelle selbst ist somit eine Ressource der Umgebung.¹⁰²⁷

Da die Agenten eine Börse bilden, können dies sowohl Präferenzen für den Kauf als auch für den Verkauf von Transportdienstleistungen sein.

6.4.2 Struktureller Aufbau

Präferenzen stellen ein Angebot für eine zusammenhängende oder eine Nachfrage nach einer zusammenhängenden Transportdienstleistung auf einer festen Strecke im zuvor beschriebenen Verkehrsnetz dar. Grundlegend ist hierbei, dass die mit Hilfe einer Präferenz zum Ausdruck gebrachten Angebote für eine und Nachfragen nach *einer* verbindungsbezogenen Transportdienstleistung meist aus mehreren Angeboten für und Nachfragen nach relationsbezogenen Teildienstleistungen bestehen.

Die folgenden Ausführungen verdeutlichen dies:

- **Anbietende Präferenzen** stellen ein Angebot für einen Transport auf einer Verbindung zwischen zwei Haltepunkten im Verkehrsnetz zu einem festen Zeitpunkt dar. Angeboten wird der Transport von Frachten einer bestimmten Güterart bis zu einem maximalen Gewicht und einem maximalen Volumen. Umfasst der Transport mehrere Relationen, so wird das Angebot derart in mehrere Teildienstleistungen aufgeteilt, dass jede Teildienstleistung genau einer Relation zugeordnet ist und die Gesamtheit der Teildienstleistungen die angebotene Transportdienstleistung in ihrer Gesamtheit umfasst.
- **Nachfragende Präferenzen** stellen eine Nachfrage nach einem Transport auf einer Verbindung zwischen zwei Haltepunkten im Verkehrsnetz zu einem festen Zeitpunkt dar. Nachgefragt wird der Transport einer Fracht einer bestimmten Güterart mit einem festen Gewicht und Volumen. Umfasst der Transport mehrere Relationen, so wird die Nachfrage derart in mehrere Teildienstleistungen aufgeteilt, dass jede Teildienstleistung genau einer Relation zugeordnet ist und die Gesamtheit der Teildienstleistungen die gesamte Nachfrage erfüllt.

Jede nachgefragte oder angebotene Teildienstleistung für einen Transport auf der Relation zwischen den Haltepunkten H_a und H_b hat die folgenden Eigenschaften:

¹⁰²⁶ Vgl. Kapitel 5.2.5, S. 83.

¹⁰²⁷ Vgl. Kapitel 5.2.5, S. 85.

- die Relation (H_a, H_b) , auf welcher der durch die Teildienstleistung erfüllte Transportabschnitt verläuft,
- das Zeitfenster (T^{H_a}, T^{H_b}) , wobei T^{H_a} den frühest möglichen Abfahrtszeitpunkt an H_a und T^{H_b} den spätest zulässigen Ankunftszeitpunkt an H_b angibt,
- die Güterart k , welche aus einer Liste an möglichen, eindeutig definierten Güterarten stammen muss,
- das Gewicht q der zu transportierenden Güter sowie
- das Volumen v der zu transportierenden Güter.

Relation, Termin und Güterart sind unveränderliche, im gesamten System eindeutige Bezeichner für Ort und Zeitpunkt des Transports sowie die zu transportierende Güterart. Die Angaben bzgl. Gewicht und Volumen beziehen sich entweder auf die zu transportierende Fracht (im Falle einer Nachfrage) oder auf die maximal verfügbare Transportkapazität (im Falle eines Angebots).

Formal kann jede Teildienstleistung als Tupel aus den beschriebenen Eigenschaften dargestellt werden:

$$g_i := \{ \underbrace{(H_a, H_b)}_{\text{Relation}}, \underbrace{(T_i^{H_a}, T_i^{H_b})}_{\text{Terminierung}}, \underbrace{k_i}_{\text{Güterart}}, \underbrace{q_i}_{\text{Gewicht}}, \underbrace{v_i}_{\text{Volumen}} \}$$

$$g_i \hat{=} \text{Teildienstleistung}$$

Eine Kombination mehrerer Teildienstleistungen ergibt ein Dienstleistungsbündel, welches als Liste dargestellt werden kann:

$$G := (g_1, g_2, \dots, g_t)$$

$$G \hat{=} \text{Gesamttransport} \qquad t \hat{=} \text{Anzahl Teildienstleistungen}$$

Eine Präferenz lässt sich als Tupel aus Dienstleistungsbündel und Zahlungsbereitschaft in Form eines Preises darstellen:

$$B_j := (G_j, p_j)$$

$B_j \hat{=}$ Gebot

$G_j \hat{=}$ Gesamttransport

$p_j \hat{=}$ Zahlungsbereitschaft

Als Differenzierungsmerkmal zwischen nachfragenden und anbietenden Präferenzen dient das Vorzeichen des Preises: Im Falle nachfragender Präferenzen wird der Preis positiv notiert, da es sich um eine Zahlungsbereitschaft für den Erhalt des Güterbündels handelt. Negativ notierte Preise signalisieren eine anbietende Präferenz, da der Anbieter mit der angegebenen Zahlungsbereitschaft für die Veräußerung des Güterbündels kompensiert werden möchte.

Die Notation der Präferenz sollte zudem ermöglichen, Alternativen zu einem Gebot in einer Weise abzugeben, welche dem Auktionsmodell signalisiert, dass maximal eines der Gebote den Zuschlag erhalten darf. Auf diese Weise können Alternativrouten angeboten werden, von denen der Anbieter im Falle eines Zuschlags maximal eine erfüllen kann.

Dies wird erreicht, indem jedem Gebot ein Alternativen-Bezeichner („ALT-Wert“) zugeordnet wird, welcher sicherstellt, dass maximal ein Gebot mit diesem ALT-Wert in der Endallokation vorkommt.¹⁰²⁸

Tabelle 5 zeigt ein Beispiel von zwei Agenten, welche fünf Gebote abgegeben haben.

Agent	Gebot	Dienstleistungsbündel	Preis	ALT-Wert
Anbieter 1	B_1	G_1	-100	1
Anbieter 1	B_2	G_2	-100	1
Anbieter 1	B_3	G_3	-120	1
Anbieter 1	B_4	G_4	-200	2
Anbieter 2	B_5	G_5	-300	3

Tabelle 5: Beispiel für Gebote¹⁰²⁹

In diesem Beispiel würde die Allokation maximal ein Gebot aus der Menge $\{B_1, B_2, B_3\}$ enthalten, da diese Gebote denselben ALT-Wert besitzen. ALT-Werte sind für jeden Agenten auf die eigenen Gebote beschränkt, da die tatsächlichen Werte vom System in einer

¹⁰²⁸ Bei SANDHOLM et al. werden derartige Bedingungen als „XOR constraints“ oder XOR bids beschrieben. Vgl. SANDHOLM et al. (2002), S. 72, sowie SANDHOLM (2002), S. 1 ff.

¹⁰²⁹ Quelle: eigene Darstellung.

Weise vergeben werden, die verhindert, dass Anbieter 1 ein ALT-Wert zugeordnet wird, der mit einem ALT-Wert von Anbieter 2 übereinstimmt.

Eine Präferenz lässt sich abschließend als Tupel aus Dienstleistungsbündel, Preis und ALT-Wert darstellen:

$$B_j := (G_j, p_j, ALT_j)$$

$B_j \hat{=}$ Gebot

$G_j \hat{=}$ Gesamttransport

$p_j \hat{=}$ Zahlungsbereitschaft

$ALT_j \hat{=}$ ALT-Wert

6.4.3 Möglichkeit der Kompatibilitätsprüfung

Da die hinterlegten Teildienstleistungen in einem einheitlichen Format notiert werden, können Angebot und Nachfrage paarweise miteinander verglichen werden. So lässt sich die Frage beantworten, ob eine angebotene Teildienstleistung eine nachgefragte Teildienstleistung erfüllen kann.

Hierzu müssen einige Bedingungen erfüllt sein:

- Identität der Haltepunkte,
- nachgefragter Zeitraum innerhalb des Angebotszeitraums,
- nachgefragte Güterart innerhalb der laut Angebot transportierbaren Güterarten,
- nachgefragtes Gewicht kleiner oder gleich dem angebotenen Maximalgewicht sowie
- nachgefragtes Volumen kleiner oder gleich dem angebotenen Maximalvolumen.

Sind alle diese Bedingungen für ein Angebot g_i und eine Nachfrage g_e erfüllt, so handelt es sich um „kompatible Präferenzen“.

6.4.4 Relevanz für die prototypische Implementierung

Die Präferenzoffenlegung findet für die Agenten aus konzeptioneller Sicht in der gemeinsamen, beobachtbaren Umgebung statt.¹⁰³⁰ Diese Umgebung ist zugänglich, da die Agenten sie vollständig wahrnehmen können. Sie ist ferner dynamisch durch die Möglichkeit der menschlichen Benutzer als Nicht-Agenten die Präferenzen eines Agenten zu verändern. Sie ist diskret, da die Ausprägungen der Präferenzen keinen kontinuierlichen Charakter haben.

¹⁰³⁰ Vgl. Kapitel 5.2.5, S. 83.

Die Kenntnis der Präferenzen anderer Agenten ist zudem Voraussetzung, um eine Kompatibilitätsprüfung durchführen zu können, die entscheidet, ob der aktuelle Agent mit den betrachteten anderen Agenten gemeinsame Handelsinteressen verfolgt. Diese Prüfung ist nötig, damit sich Agenten mit kompatiblen Handelsinteressen zu Gruppen zusammenschließen können.

Aus Sicht der BDI-Architektur hat ein Agent nach Austausch der Präferenzen die eigenen Kontakte und die Präferenzen seiner bekannten Kontakte als Ansichten („beliefs“), seine eigenen Präferenzen als Wünsche („desires“) sowie seine Strategien zur Gruppenbildung als Intentionen („intentions“).¹⁰³¹

Sie stellen eine Bietsprache im Rahmen der durchzuführenden Auktionen dar und bilden die Grundlage der zu konzipierenden Spezifikationssprache für die Agenten.¹⁰³²

6.5 Gruppenbildung

6.5.1 Notwendigkeit der Gruppenbildung im dezentralen Netzwerk

Nachdem alle Agenten ihre Präferenzen untereinander ausgetauscht haben, stehen sie vor dem Koordinationsproblem, dass sie aufgrund des fehlenden zentralen Koordinators neben dem Auffinden potentieller Handelspartner auch die Anbahnung und Durchführung von Auktionen untereinander koordinieren müssen. Um im dezentralen Netzwerk Auktionen durchführen zu können, bei denen alle Auktionsteilnehmer auch tatsächlich eine Präferenz für die versteigerten Güter besitzen, müssen sich die Agenten mit kompatiblen Präferenzen zu *Gruppen* zusammenfinden und dynamisch einen solchen Koordinator aus ihrer Mitte wählen. Dieser Agent wird im Folgenden der *Moderator* der Gruppe genannt.¹⁰³³

Die Gruppen stellen soziale Strukturen für die Agenten dar.¹⁰³⁴

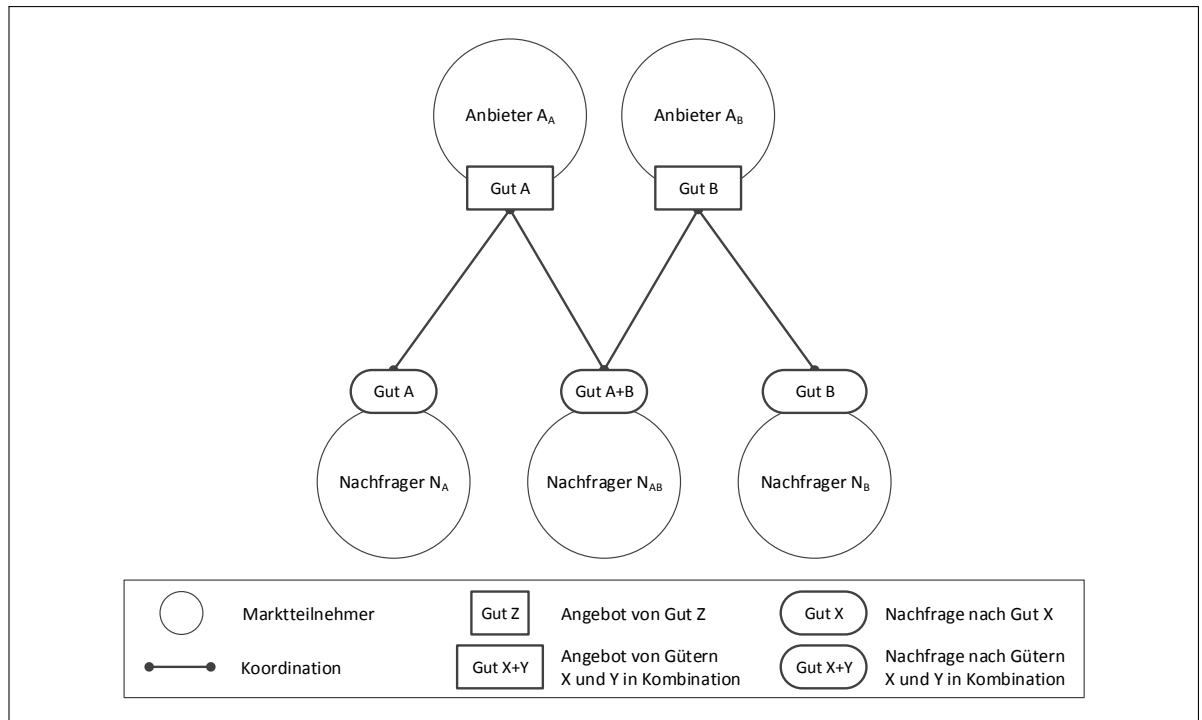
Das folgende Beispiel in Abbildung 27 illustriert diese Notwendigkeit:

¹⁰³¹ Vgl. Kapitel 5.2.3, S. 78.

¹⁰³² Vgl. Kapitel 5.3.3.4, S. 105, sowie Kapitel 5.2.3, S. 79.

¹⁰³³ Der Begriff „Moderator“ wird gewählt, da der in diese Position gewählte Agent in den verschiedenen Gruppenphasen sowohl die Rolle des Koordinators als auch des Auktionators einnimmt. Die einzelnen Phasen werden im Folgenden noch beschrieben.

¹⁰³⁴ Vgl. Kapitel 5.2.5, S. 84.

Abbildung 27: Einfaches Beispiel zur Gruppenbildung¹⁰³⁵

1. Nachfrager N_A möchte Gut A kaufen.
2. Anbieter A_A möchte Gut A verkaufen.
3. Nachfrager N_B möchte Gut B kaufen.
4. Anbieter A_B möchte Gut B verkaufen.
5. Nachfrager N_{AB} möchte Gut A und B als Kombination kaufen.

Das zu implementierende Multi-Agenten-System muss in der Lage sein, zu erkennen, dass A_A und A_B gemeinsam ihre Güter A und B in einer kombinatorischen Auktion versteigern müssen, da N_{AB} Güter von beiden nachfragt. Ob die Präferenz von N_{AB} hierbei stärker ist als die addierten Präferenzen von N_A und N_B und welcher der drei Nachfrager schließlich den Zuschlag erhält, ist für diese Betrachtung irrelevant. Entscheidend ist, dass der Gruppenbildungsprozess berücksichtigt, welche Angebote zusammengefasst zu den gebündelten Nachfragen der Teilnehmer passen.

Weiterhin müssen Gruppen verschiedene Phasen durchlaufen:

- eine anfängliche Phase, in der die Gruppe offen für den Beitritt neuer Mitglieder ist,
- eine die Auktion vorbereitende Phase, in der nur noch Agenten beitreten können, welche die in der Gruppe versteigerten Güter nachfragen,
- eine Phase, in der keine Agenten mehr beitreten können und die Auktion durchgeführt wird, sowie

¹⁰³⁵ Quelle: eigene Darstellung.

- eine Phase, in der sich die Agenten auf die durch den Auktionator festgelegte Endallokation der Güter einigen.

Diese Gruppenphasen sind in Anlehnung an das im Rahmen elektronischer Marktplätze besprochene Transaktionsphasenmodell aus Kapitel 5.1.5 modelliert.¹⁰³⁶

Haben die designierten Mitglieder einer Gruppe sich noch nicht zusammengefunden, so befinden sie sich noch in der Informationsphase. Es werden Informationen zu möglichen Transaktionen in Form von Präferenzen ausgetauscht.

Die erste und zweite Gruppenphase lehnen sich an die Vereinbarungsphase an, in welcher potentielle Käufer und Verkäufer einander eine Transaktionsabsicht signalisieren. Es kommt zu der Vereinbarung, gemeinsam eine Auktion durchzuführen.

Die dritte und vierte Gruppenphase sind nach dem Vorbild der Abwicklungsphase konzipiert, in welcher die Teilnehmer gegenseitig die von ihnen beschriebenen Leistungen erbringen. Die im Rahmen der Gruppe beschriebene Leistung ist stets die Durchführung der Auktion und die anschließende Einigung auf die Allokation der Güter.

Ferner müssen Gruppen von den Agenten intern als inaktiv gekennzeichnet werden können, wenn sie die grade skizzierten Phasen bereits durchlaufen haben oder aus anderen Gründen für gruppenspezifische Prozesse nicht länger zur Verfügung stehen.¹⁰³⁷

Aus diesen Überlegungen folgt, dass jede Gruppe die folgenden Eigenschaften besitzen muss:¹⁰³⁸

- eindeutige Bezeichnung,
- Liste der zu versteigernden Güter,
- Liste der Mitglieder,
- Status sowie
- Moderator.

Jede Gruppe benötigt eine systemweit eindeutige Bezeichnung, um bspw. Einladungen in eine Gruppe kommunizieren zu können.

Die Liste der zu versteigernden Güter entspricht Repräsentationen der von den Agenten angebotenen oder nachgefragten Transportdienstleistungen.

Die Liste der Mitglieder enthält die Bezeichner und Adressen der in der Gruppe registrierten Agenten.

¹⁰³⁶ Vgl. Kapitel 5.1.5, S. 70.

¹⁰³⁷ Gemeint sind hier die im folgenden Kapitel beschriebenen Prozesse zur Gruppenbildung und Gruppenzusammenlegung. Vgl. Kapitel 6.5.2, S. 141.

¹⁰³⁸ Eine technische Beschreibung der Implementierung dieser Eigenschaften findet sich im Anhang. Vgl. Anhang, S. 313.

Der Status einer Gruppe gibt an, ob die Gruppe noch aktiv ist und, wenn sie noch aktiv ist, in welcher Phase sich die Gruppe befindet.

Zudem muss jeder Gruppe ein Moderator vorstehen. Diese Rolle fällt zunächst dem Gründer der Gruppe zu.

6.5.2 Vorgehensweise

6.5.2.1 Gründung von Gruppen

In einer BDI-Architektur folgen die Aktivitäten von Agenten den Konzepten von Zielermäßigung und Abwägung, welche zur Aktivierung von Plänen führen. Die im Folgenden beschriebenen Pläne stellen verschiedene Wege zur Erreichung desselben Ziels – der Durchführung einer Auktion – dar, so dass der Agent abhängig von seinen Beobachtungen der Umgebung zwischen verschiedenen Plänen zur Erreichung seines Ziels abwägen kann.¹⁰³⁹

Nachdem ein Agent mit einer Konfiguration aus Präferenzen gestartet wurde, teilt er diese Präferenzen mit seinem Netzwerk. Alle Agenten kennen die Präferenzen aller anderen Agenten in ihrem Netzwerk und die Präferenzen liegen in einem einheitlichen Format vor. Das Ziel eines Agenten besteht nun darin, die Transportdienstleistungen zu erwerben, die sein menschlicher Benutzer nachfragt, und jene Transportdienstleistungen zu veräußern, die sein menschlicher Benutzer anbietet. Aus Gründen der Lesbarkeit werden im Folgenden Agenten, die stellvertretend für ihre menschlichen Benutzer Transportdienstleistungen anbieten, als „Anbieter“ und solche, die stellvertretend Transportdienstleistungen nachfragen, als „Nachfrager“ bezeichnet.

Anbieter gründen und moderieren Gruppen mit dem Ziel, ihre angebotenen Transportdienstleistungen als Güter zu versteigern.

Jeder anbietende Agent erstellt zu diesem Zweck eine Liste von Kandidaten aus allen Nachfragern in seinem Netzwerk. Er prüft innerhalb dieser Liste für jeden Nachfrager N , ob seine angebotenen Güter einen Teil der Präferenzen von N abdecken. Trifft dies zu, überprüft der Anbieter, ob zur Abdeckung der ermittelten Präferenzen eines Nachfragers N weitere Güter angeboten werden müssen, die der Anbieter nicht anbietet. Kann er mit seinem Angebot die Nachfrage von N nur teilweise erfüllen, so identifiziert der Anbieter weitere relevante Anbieter aus seinem Netzwerk, die mit ihren angebotenen Gütern dazu beitragen, die Nachfrage von N möglichst vollständig zu erfüllen.

Nachdem neben dem Nachfrager N weitere relevante Anbieter identifiziert wurden, lädt der Anbieter den betreffenden Nachfrager und die relevanten weiteren Anbieter in eine

¹⁰³⁹ Vgl. Kapitel 5.2.6, S. 87.

Gruppe zur Versteigerung der in den Präferenzen beschriebenen Transportdienstleistungen ein. Sollte noch keine Gruppe existieren, in der die betreffenden Transportdienstleistungen als Güter versteigert werden, so gründet der Anbieter zu diesem Zweck eine neue Gruppe.

Dieses Verfahren lässt sich anhand des Beispiels aus Abbildung 27 darstellen:

1. Anbieter A_A möchte Gut A verkaufen.
2. Anbieter A_B möchte Gut B verkaufen.
3. Die Nachfrager N_A und N_B möchten Gut A bzw. B jeweils einzeln kaufen.
4. Der Nachfrager N_{AB} möchte Gut A und B als Kombination kaufen.

Im Rahmen des beschriebenen Verfahrens möchte Anbieter A_A Gut A an die Nachfrager N_A und N_{AB} verkaufen. Er stellt fest, dass der Nachfrager N_{AB} neben dem Gut A auch B nachfragt, und erkennt, dass er selbst dieses Gut zwar nicht anbietet, er allerdings mit A_B einen Agenten in seinem Netzwerk kennt, der diese Nachfrage zusammen mit ihm vollständig erfüllen könnte. Anbieter A_A gründet eine Gruppe zur Versteigerung von A und B und lädt sowohl N_{AB} als auch A_B ein.

Dieser Prozess wird von jedem Agenten autonom mit einer in seiner Software vorgegebenen Regelmäßigkeit ausgeführt.

6.5.2.2 Einladungen zu existierenden Gruppen

Die Moderatoren bereits existierender Gruppen prüfen ihre beobachtbare Umgebung regelmäßig auf weitere potentielle Handelspartner. Dies können Nachfrager sein, welche die in der Gruppe versteigerten Güter nachfragen, oder Anbieter, welche nur teilweise erfüllte Präferenzen der Nachfrager in der Gruppe derart ergänzen, dass diese zu einem höheren Grad erfüllt werden.

Bei der Einladung zu einer existierenden Gruppe ändern sich die innerhalb der Gruppe versteigerten Güter nicht.

6.5.2.3 Zusammenschluss von existierenden Gruppen

Ferner versuchen die Moderatoren von Gruppen stets andere Gruppen zu identifizieren, mit denen sie sich zusammenschließen können, um die Nachfrage der jeweiligen Teilnehmer zu einem höheren Grad zu erfüllen.

Hierzu identifiziert der Moderator einer Gruppe in allen anderen Gruppen, in denen er Mitglied ist oder die er selbst moderiert, jene Kandidaten, die für einen Zusammenschluss in Frage kommen.

Die hierbei angewendeten Kriterien sind:

1. die Gruppe wird ebenfalls vom Moderator moderiert
oder
der Moderator wurde in seiner Rolle als Anbieter in die betreffende Gruppe eingeladen und fragt keine Güter in der Gruppe nach und
2. die Gruppe befindet sich im anfänglichen Status, so dass weitere Mitglieder beitreten können.

Alle Gruppen, die beide Kriterien erfüllen, bilden die Liste möglicher Kandidaten.

Der Moderator vergleicht nun in einem iterativen Verfahren jede Gruppe in der Liste möglicher Kandidaten mit jeder anderen Gruppe. Dabei wird für jede Kandidatengruppe K überprüft, ob sich unter allen anderen Gruppen eine Gruppe M findet, deren nachfragende Mitglieder an zumindest einem der Güter der Kandidatengruppe K interessiert sind. Fällt diese Prüfung positiv aus, wird dem Moderator der Kandidatengruppe K eine Nachricht geschickt, welche einen Zusammenschluss der Gruppen K und M vorschlägt. Der Moderator der Kandidatengruppe K stimmt diesem Vorschlag mit einer entsprechenden Antwort-Nachricht zu und die Gruppen K und M werden zu einer neuen Gruppe Z zusammengeschlossen.

Beim Zusammenschluss zweier Gruppen wird eine neue Gruppe gegründet, die

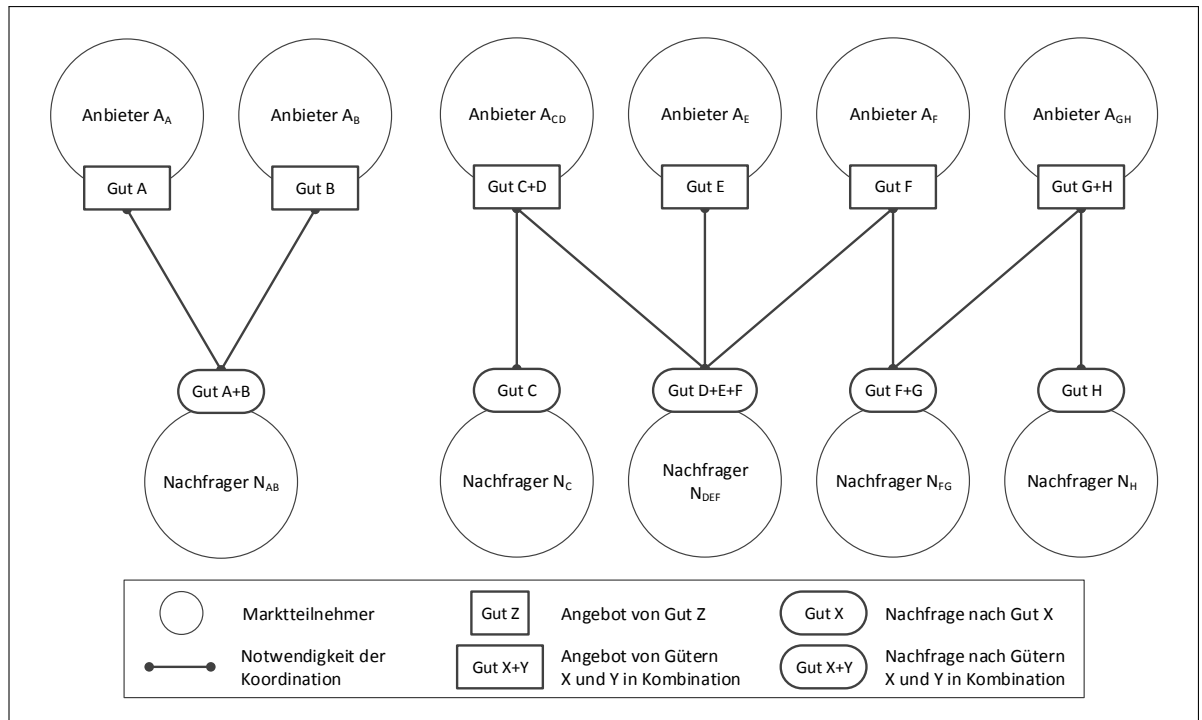
- eine neue Bezeichnung erhält,
- den Moderator der nach Mitgliedern größeren der beiden ursprünglichen Gruppen als Moderator bestimmt¹⁰⁴⁰ und
- sich aus den Mitgliedern der ursprünglichen Gruppen und jenem Moderator, der nicht Moderator der neuen Gruppe geworden ist, zusammensetzt.

Beim Zusammenschluss von existierenden Gruppen wird innerhalb der neuen Gruppe die Gesamtmenge aller Güter der beiden zusammengeschlossenen Gruppen versteigert.

Da zu den Kandidatengruppen auch vom Moderator selbst moderierte Gruppen gehören, kann es sein, dass dieser Prozess ergibt, dass zwei von ihm moderierte Gruppen zusammengeschlossen werden können. In diesem Fall werden die Gruppen ohne Versenden einer Nachricht direkt zusammengeschlossen.

Das Beispiel in Abbildung 28 illustriert diese Notwendigkeit:

¹⁰⁴⁰ Sind die beiden ursprünglichen Gruppen nach Mitgliedern gleich groß, so bestimmt sich die Größe nach der Anzahl der versteigerten Güter. Ist auch diese für beide Gruppen identisch, so wird der Moderator der länger existierenden Gruppe als neuer Moderator bestimmt.

Abbildung 28: Erweitertes Beispiel zur Gruppenbildung¹⁰⁴¹

Anbieter und Nachfrager bilden Gruppen, wie zuvor beschrieben. Ohne die in diesem Kapitel beschriebenen Verfahren würden bspw. die folgenden drei Gruppen entstehen:

- Gruppe 1: A_A, A_B mit N_{AB}
- Gruppe 2: A_{CD}, A_E, A_F mit N_C, N_{DEF}, N_{FG}
- Gruppe 3: A_{GH}, A_F mit N_{DEF}, N_{FG}

Unter Berücksichtigung der in den vorangegangenen Kapiteln beschriebenen Verfahren wären die Agenten nicht in der Lage zu erkennen, dass sich Anbieter A_F in zwei Gruppen befindet, die er nicht moderiert, die jedoch von einem Zusammenschluss profitieren würden. Das beschriebene Verfahren würde in diesem Beispiel einen Zusammenschluss herbeiführen, da Anbieter A_F Mitglied in den Gruppen 2 und 3 ist sowie Nachfrager N_{DEF} und N_{FG} Güter aus beiden Gruppen nachfragen.

Ein weiterer Effekt des beschriebenen Verfahrens ist die Konsolidierung von Gruppen. In dem in Abbildung 27 gegebenen Beispiel führt der Anbieter A_A das beschriebene Verfahren aus und gründet eine eigene Gruppe, zu der er Anbieter A_B einlädt. Unter Umständen hat der Anbieter A_B bereits eine Gruppe gegründet, um Gut B an den Nachfrager N_B zu versteigern.

Das in diesem Kapitel beschriebene Verfahren sorgt in dieser Konstellation dafür, dass die entstandenen beiden Gruppen zusammengeschlossen und so die Präferenzen möglichst vieler Nachfrager möglichst vollständig erfüllt werden.

¹⁰⁴¹ Quelle: eigene Darstellung.

6.5.2.4 Vorschläge für den Zusammenschluss von Gruppen durch Mitglieder

Neben den im vorherigen Kapitel beschriebenen Verfahren, bei denen die Moderatoren den Vorschlag eines Zusammenschlusses zweier Gruppen unterbreiten, sind noch zwei weitere Verfahren notwendig, um die gewünschte Gruppenbildung zu erzielen.

Bei beiden Verfahren wird abermals über eine Liste von Kandidatengruppen iteriert. Diese umfasst diesmal alle Gruppen, in denen der Agent nicht selbst Moderator, sondern lediglich Mitglied ist.

Beide Verfahren werden nacheinander auf die Liste der Kandidatengruppen angewandt und verlaufen wie folgt:

1. Der Agent prüft für jede Kandidatengruppe K , ob er ein Gut k nachfragt, das in dieser Gruppe versteigert wird. Trifft dies zu und ist k Teil eines Güterbündels, so prüft er, ob er Mitglied einer weiteren Gruppe M ist, in der Güter versteigert werden, die er als Teil desselben Güterbündels nachfragt und die nicht in der Gruppe K versteigert werden.
2. Der Agent prüft für jede Kandidatengruppe K , ob in ihr eine Nachfrage für ein von ihm angebotenes Gut k besteht, das jedoch nicht Teil der versteigerten Güter in der Kandidatengruppe K ist. Trifft dies zu, so prüft er, ob er Mitglied einer weiteren Gruppe M ist, in der k Teil der versteigerten Güter ist und er als Anbieter von k auftritt.

Natürlichsprachlich ausgedrückt sucht der Agent für jeden Fall einer teilweise erfüllten Präferenz nach Gruppen, in denen weitere Teile dieser Präferenz erfüllt werden. Wird er bei dieser Suche fündig, schlägt er den Moderatoren der jeweiligen Gruppen einen Zusammenschluss vor.

6.5.3 Relevanz für die prototypische Implementierung

Die Gruppenbildung erfüllt eine wichtige Funktion im Rahmen der Anbahnung von Auktionen: Sie bringt Agenten mit kompatiblen Präferenzen in eine organisierte Struktur, welche gruppeninterne und -externe Kommunikation und Koordination ermöglicht. Das Ziel ist die Einbeziehung möglichst vieler Agenten mit kompatiblen Präferenzen, um auf diesem Wege einen möglichst hohen Wettbewerb um die in der Gruppe versteigerten Güter herbeizuführen.

Die Gruppe ist durch die vordefinierten Handlungen einzelner Agenten innerhalb und außerhalb der Gruppe, wie bspw. die beschriebenen Einladungen an gruppenexterne Agenten und die Möglichkeit der Koordination von Gruppenzusammenschlüssen auf Vorschlag, zu einem nach außen kohärent und individuell erscheinenden Gruppenverhalten fähig. Die

Agenten einer Gruppe schließen sich so zu einem koordiniert agierenden Multi-Agenten-System zusammen.¹⁰⁴²

Die gebildeten Gruppen können von allen Agenten wahrgenommen werden. Durch ihre beobachtbaren Eigenschaften und ihre Instrumentalität bei der Anbahnung von Auktionen stellen sie aus konzeptioneller Sicht sowohl Ressourcen als auch Funktionalitäten in der Umgebung der Agenten dar.¹⁰⁴³ Durch die Bildung der Gruppen wird die Umgebung der Agenten zudem deterministisch, da die Zustände jener Gruppen, an denen ein Agent beteiligt ist, seine zukünftigen Handlungen beeinflussen können. Aus denselben Gründen ist die Umgebung nicht episodisch.¹⁰⁴⁴

Die Organisation der Agenten innerhalb von Gruppen ermöglicht im nächsten Schritt eine Versteigerung der angebotenen Güter im Kreis der Gruppenmitglieder und des Moderators. Durch die Bildung der Gruppen auf Basis von kompatiblen Handelspräferenzen wird die strategische Komplexität der folgenden Auktion gemindert.¹⁰⁴⁵ Zudem begünstigt die beschriebene Vorgehensweise Strategiebeständigkeit und bietet Anreizkompatibilität für die Teilnehmer der Auktion.¹⁰⁴⁶

6.6 Ad-hoc-Auktionen

6.6.1 Begriff der Ad-hoc-Auktion

Das letzte verbleibende organisationale Problem bei der Konzeption des AFEX-Systems besteht in der Prämisse, dass einem dezentral organisierten Marktplatz neben der Figur des zentralen Betreibers auch die eines zentralen Auktionators fehlt. Aus diesem Grund müssen die Agenten sich nicht nur selbstständig im Internet zusammenfinden, sondern auch die Anbahnung und Durchführung von Auktionen untereinander koordinieren. Die Agenten gleichen ihre Handelsinteressen in Form von Präferenzen miteinander ab und bilden zum Zweck der Auktionsdurchführung, wie im vorangegangenen Kapitel beschrieben, Gruppen. Der entscheidende Unterschied zu zentral organisierten Marktplätzen ist, dass es keine unabhängige, zentrale Auktionator-Instanz gibt, sondern der Moderator einer Gruppe gleichzeitig die Rolle des Auktionators übernimmt.

Auktionen, welche ohne Rückgriff auf strukturelle Bedingungen in dieser Weise angebahnt und durchgeführt werden, werden aufgrund ihres spontan autonomen Charakters im Rahmen dieser Forschungsarbeit als *Ad-hoc-Auktionen* bezeichnet.

¹⁰⁴² Vgl. Kapitel 5.2.7, S. 90.

¹⁰⁴³ Vgl. Kapitel 5.2.5, S. 85.

¹⁰⁴⁴ Vgl. Kapitel 5.2.5, S. 83.

¹⁰⁴⁵ Vgl. Kapitel 5.3.3.4, S. 103.

¹⁰⁴⁶ Vgl. Kapitel 5.3.1, S. 97.

Abbildung 29 zeigt eine solche Auktion, welche von den Teilnehmern autonom organisiert wurde.

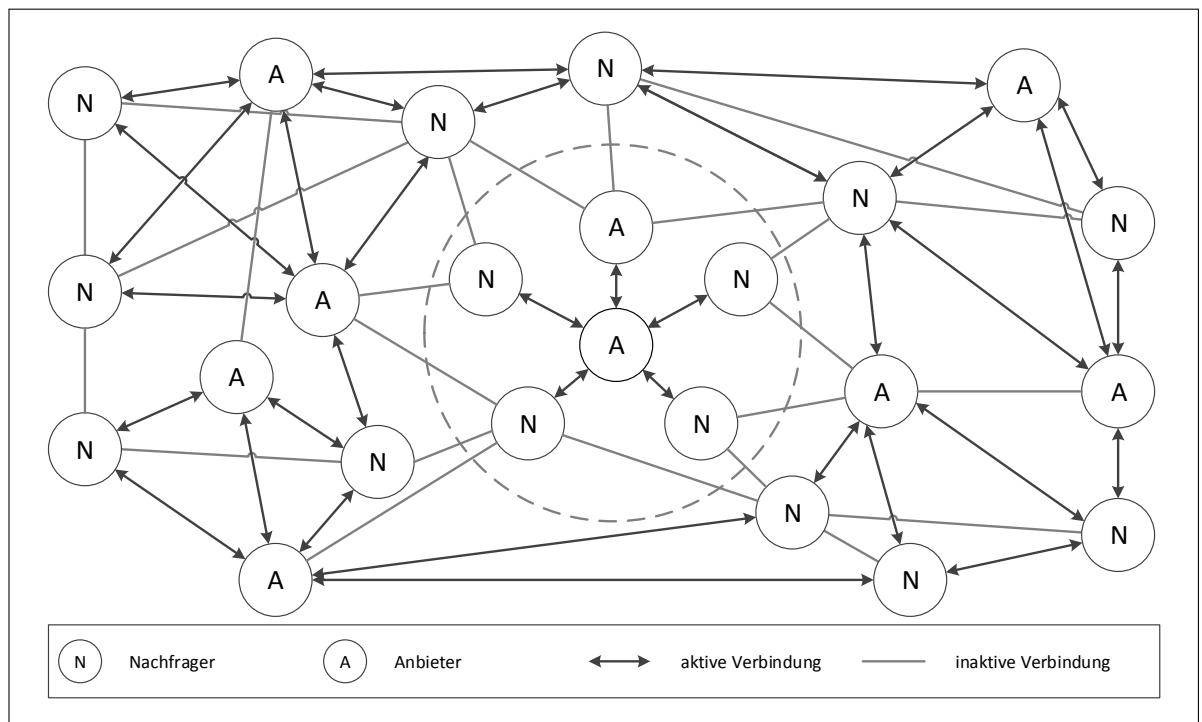


Abbildung 29: Beispiel für eine Ad-hoc-Auktion¹⁰⁴⁷

Die 25 Teilnehmer des Marktes sind alle untereinander über aktive und inaktive Verbindungen vernetzt. Sie tauschen Kontakt- und Präferenzdaten miteinander aus. Im Zentrum der Darstellung haben sich sechs Teilnehmer zusammengefunden, die ein gemeinsames Interesse an verschiedenen Gütern (Transportdienstleistungen) haben und eine zweiseitige, kombinatorische Auktion durchführen (in Abbildung 29 durch eine gestrichelte Ellipse eingerahmt).

Während der Auktion halten alle Agenten weiter Kontakt zu anderen Agenten, nehmen jedoch nicht an weiteren Auktionen zu anderen Gütern teil.

6.6.2 Begriff der Vollständigkeit von Gruppen

Wenn die versteigerten Güter in einer Gruppe mit den aktuell in dieser Gruppe vertretenen Agenten (Moderator und Mitglieder) versteigert werden können, so wird die Gruppe als „anbieterseitig vollständig“ bezeichnet. Dies bedeutet, dass die Nachfrager in einer Gruppe zusammen alle versteigerten Güter nachfragen. Die Nachfragen bestehen nur in Kombinationen, deren Elemente ausschließlich aus der Menge der versteigerten Güter stammen.

¹⁰⁴⁷ Quelle: eigene Darstellung.

Es soll an dieser Stelle betont werden, dass die Nachfrager durchaus Präferenzen neben den versteigerten Gütern der Gruppe haben können, solange diese Präferenzen nicht kombinatorisch mit Präferenzen für die versteigerten Güter der Gruppe verknüpft sind.

Abbildung 30 zeigt drei Gruppen, von denen nur Gruppe 3 anbieterseitig vollständig ist.

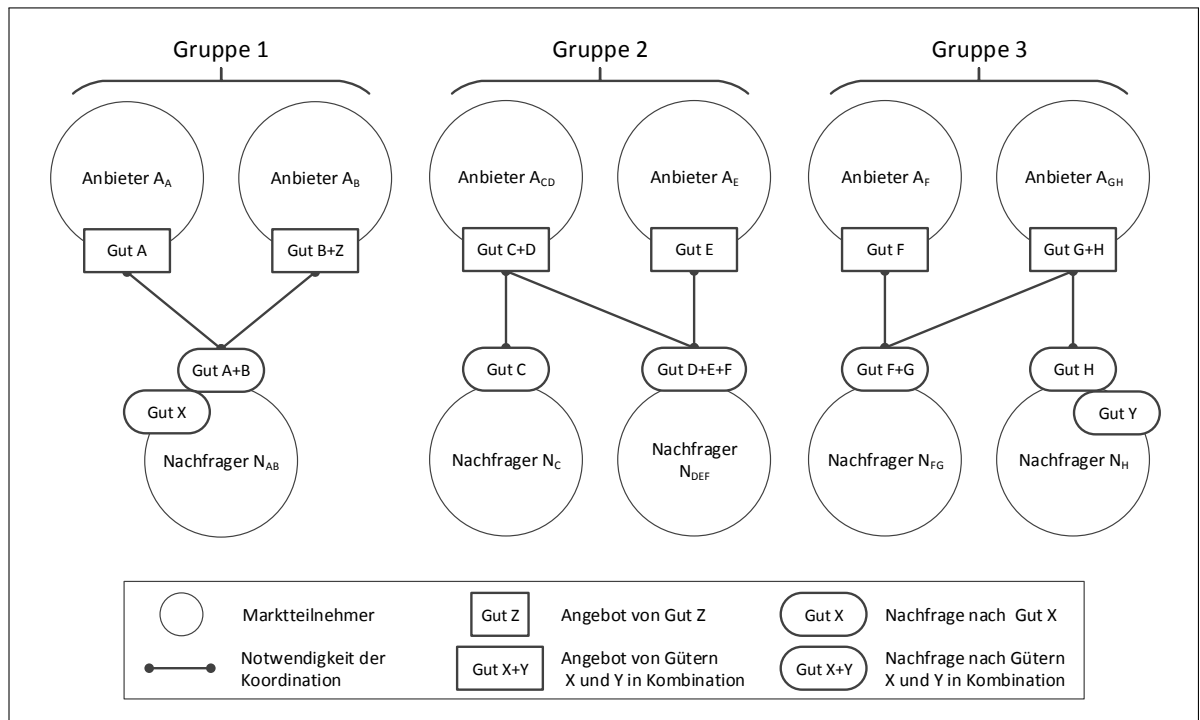


Abbildung 30: Beispiel zur Vollständigkeit von Gruppen¹⁰⁴⁸

In Gruppe 1 werden die Güter B und Z in Kombination angeboten, was eine Versteigerung an Nachfrager N_{AB} verhindert. Die Tatsache, dass der Nachfrager N_{AB} unabhängig von den Gütern A und B auch X nachfragt, berührt diesen Sachverhalt nicht.

In Gruppe 2 werden die Güter C , D und E versteigert und die Güter C , D , E und F nachgefragt. Da die Güter D , E und F in Kombination nachgefragt werden und somit keine Versteigerung möglich ist, ist die Gruppe nicht anbieterseitig vollständig.

In Gruppe 3 werden die Güter F , G und H versteigert und auch nachgefragt. Die Gruppe ist anbieterseitig vollständig. Die Tatsache, dass der Nachfrager N_H auch das Gut Y nachfragt, berührt diesen Sachverhalt nicht.

6.6.3 Anbahnung von Ad-hoc-Auktionen

Die in Kapitel 6.5 beschriebene Gruppenbildung dient der Anbahnung von Ad-hoc-Auktionen. Die gebildeten Gruppen zeichnen sich durch kompatible Präferenzen aus. Zur

¹⁰⁴⁸ Quelle: eigene Darstellung.

Anbahnung einer Ad-hoc-Auktion ist es nötig, dass die betreffende Gruppe anbieterseitig vollständig ist.

6.6.4 Gruppenphasen im Rahmen von Ad-hoc-Auktionen

6.6.4.1 Überblick

Eine Gruppe befindet sich zu jedem Zeitpunkt in einer Phase. Jede Gruppe kann im Rahmen der Anbahnung und Durchführung einer Ad-hoc-Auktion sechs Phasen durchlaufen.

Tabelle 6 zeigt eine Übersicht der vier aktiven Phasen und der beiden möglichen Abschlussphasen.

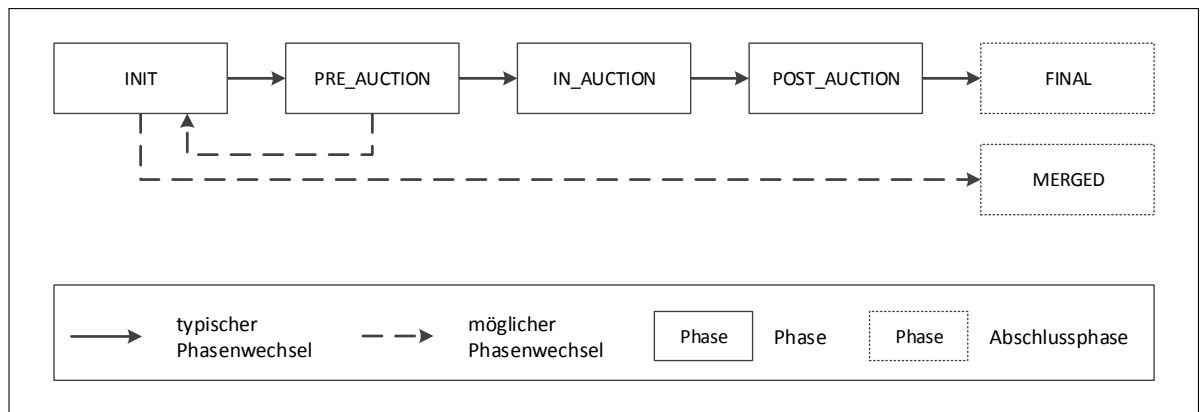
Phasenbezeichner	Beschreibung
INIT	initiale Phase, in der jede Gruppe startet
PRE_AUCTION	Vorrunde zur Auktion
IN_AUCTION	die eigentliche Auktion findet statt
POST_AUCTION	die Ergebnisse der Auktion werden kommuniziert
FINAL	Abschlussphase nach erfolgreicher Auktion
MERGED	Abschlussphase nach Gruppenzusammenführung

Tabelle 6: Übersicht über die Gruppenphasen im Rahmen von Ad-hoc-Auktionen¹⁰⁴⁹

Die ersten vier Phasen werden in der Regel sequentiell durchlaufen, bevor die Gruppe in der Abschlussphase „FINAL“ endet. Die Ausnahmen von dieser Regel sind, dass eine Gruppe von der „PRE_AUCTION“-Phase zurück in die „INIT“-Phase und von der „INIT“-Phase direkt in die Abschlussphase „MERGED“ wechseln kann. Der Unterschied zwischen aktiven Phasen und Abschlussphasen besteht darin, dass auf eine Abschlussphase keine weitere Phase folgt.

Abbildung 31 verdeutlicht diesen Sachverhalt.

¹⁰⁴⁹ Quelle: eigene Darstellung.

Abbildung 31: Gruppenphasen im Rahmen von Ad-hoc-Auktionen¹⁰⁵⁰

Im Folgenden werden die einzelnen Phasen, die Übergänge zwischen den Phasen sowie ihr Zweck im Rahmen der Auktionsdurchführung ausführlicher beschrieben.

6.6.4.2 INIT-Phase

Jede Gruppe startet in der Phase „INIT“.

In dieser Phase können externe Agenten auf Einladung in die Gruppe eintreten. Dies geschieht, wie zuvor beschrieben, zu dem Zweck, die Bandbreite der in der Gruppe angebotenen Güter zu erhöhen. Zum gleichen Zweck kann sich die Gruppe mit anderen Gruppen zusammenschließen, die sich ebenfalls in der INIT-Phase befinden.

In der INIT-Phase wird regelmäßig geprüft, ob eine Versteigerung der angebotenen Güter möglich ist, d.h., ob die Gruppe „anbieterseitig vollständig“ ist. Sobald dies der Fall ist, startet ein gruppeninterner Zeitgeber einen Countdown von 30 Sekunden. Dieser Wert wurde im Rahmen der Implementierung der Software-Agenten bewusst niedrig gewählt, um schnellere Entwicklungszyklen zu ermöglichen. Für einen späteren Produktionsbetrieb der Online-Frachtenbörse würde dieser Wert um ein vielfaches höher liegen, um Agenten den Beitritt zur Gruppe offenzuhalten.

Ist der Countdown abgelaufen, wechselt die Gruppe in die Phase „PRE_AUCTION“ und die Auktion beginnt. Verlassen anbietende Agenten die Gruppe, bevor der Countdown abgelaufen ist, wird der Countdown vom Zeitgeber zurückgesetzt und die Prüfung der Vollständigkeit der Gruppe erfolgt von neuem.

¹⁰⁵⁰ Quelle: eigene Darstellung.

6.6.4.3 PRE_AUCTION-Phase

Mit Erreichen der „PRE_AUCTION“-Phase wird erneut ein Countdown von 30 Sekunden gestartet. Auch dieser Wert ist mit Hinblick auf die leichtere Durchführung einer Vielzahl an Simulationen bewusst niedrig gewählt.

Der Gruppe können in dieser Phase immer noch Agenten beitreten, aber die Gruppe wird nicht mehr an Zusammenschlüssen teilnehmen. Sollten anbietende Agenten die Gruppe verlassen, wird der Countdown zurückgesetzt. Die Gruppe kehrt in diesem Fall in die „INIT“-Phase zurück.

Ist der Countdown abgelaufen, ohne dass Agenten die Gruppe verlassen haben, wechselt die Gruppe in die Phase „IN_AUCTION“.

6.6.4.4 IN_AUCTION-Phase

Während der Phase „IN_AUCTION“ führt der Moderator der Gruppe die Auktion der angebotenen Güter durch.

Er bedient sich hierbei der ihm vorliegenden Präferenzen, um die Gebote jedes Teilnehmers (einschließlich seiner selbst) zu formulieren. Anschließend errechnet er mit Hilfe des Auktionsmodells eine Allokation der versteigerten Güter.

Ist die Auktion durchgeführt, wechselt die Gruppe in die Phase „POST_AUCTION“.

6.6.4.5 POST_AUCTION-Phase

Zu Beginn der Phase „POST_AUCTION“ teilt der Moderator der Gruppe das Ergebnis der Auktion in Form der Allokation der versteigerten Güter mit.

Nachdem alle Mitglieder der Gruppe dieses Ergebnis bestätigt haben, wechselt die Gruppe in die Phase „FINAL“. Die Auktion ist beendet.

6.6.4.6 FINAL-Phase

Die Abschlussphase „FINAL“ wird erreicht, wenn in der Gruppe eine Auktion durchgeführt und wie oben beschrieben beendet wurde.

Die Gruppe gilt in dieser Phase als nicht mehr aktiv.

6.6.4.7 MERGED-Phase

Die Abschlussphase „MERGED“ wird erreicht, wenn die betreffende Gruppe mit einer anderen Gruppe zusammengeführt wurde. In diesem Fall wird aus den beiden zusammenzuführenden Gruppen eine neue Gruppe (mit der Phase „INIT“) gebildet und die beiden ursprünglichen Gruppen werden in die „MERGED“-Phase überführt.

Die ursprünglichen Gruppen gelten in dieser Phase als nicht mehr aktiv.

6.6.5 Transportdienstleistungen als Güter in Auktionen

Im AFEX-System sollen Angebote von und Nachfragen nach Transportdienstleistungen versteigert werden. Diese können, wie in Kapitel 6.4.2 dargestellt, im Rahmen von Präferenzen als Tupel aus Dienstleistungsbündel, ALT-Wert und Preis notiert werden.¹⁰⁵¹ Das Dienstleistungsbündel stellt eine Kombination aus nachgefragten oder angebotenen Teildienstleistungen dar, welche in ihrer Gesamtheit nötig sind, um eine Transportdienstleistung zu erbringen.

Da das Dienstleistungsbündel eine Kombination aus versteigerbaren Gütern im Sinne einer kombinatorischen Auktion darstellt, werden die in einem Dienstleistungsbündel befindlichen Teildienstleistungen im Rahmen von Auktionen als „Güter“ bezeichnet und die Dienstleistungsbündel als „Güterbündel“.¹⁰⁵²

Sobald sich Anbieter und Nachfrager mit kompatiblen Präferenzen in einer Gruppe zum Zweck der Auktion bestimmter Güter zusammengefunden haben, ist diese Tatsache jedoch ohne Bedeutung. Grund hierfür ist, dass die betrachteten Dimensionen innerhalb des zweiseitigen Auktionsmodells der Preis, die Kombination sowie der ALT-Wert sind. Ob es sich bei dem „Gut“ um eine Transport- oder Personaldienstleistung oder ein physisches Gut handelt, spielt zur Ermittlung der Gewinner der Auktion keine Rolle. Jedes Gut muss lediglich eine eindeutige Bezeichnung innerhalb der Auktion haben, so dass die Teilnehmer der Auktion in der Lage sind, ihre Präferenzen für das Gut zum Ausdruck zu bringen.

Zur Durchführung der Auktion wird das allgemeine Auktionsmodell für kombinatorische Börsen nach SANDHOLM et al. verwendet, welches zuvor in Kapitel 5.3.3.6 besprochen wurde.¹⁰⁵³

¹⁰⁵¹ Vgl. Kapitel 6.4.2, S. 134.

¹⁰⁵² Der Begriff „Güter“ bezeichnet im Kontext von Auktionen im Folgenden die versteigerten Teildienstleistungen. Sollten mit dem Begriff „Güter“ die physisch im Rahmen der Teildienstleistungen transportierten oder zu transportierenden Güter gemeint sein, so wird dies explizit benannt.

¹⁰⁵³ Vgl. Kapitel 5.3.3.6, S. 108.

Als Zielfunktion des Auktionsmodells wird „Ertragsmaximierung“ gewählt.¹⁰⁵⁴ Die Gebotsabgabe erfolgt verdeckt an den Auktionator, welcher die Allokation und damit die Gewinner der Auktion ermittelt. Die Auktion umfasst eine Runde.

Mit Hinblick auf die in Kapitel 5.3.3.4 vorgestellte Differenzierung in offene und geschlossene Auktionsmodelle sind Ad-hoc-Auktionen eine Hybridform:¹⁰⁵⁵ Einerseits handelt es sich um eine geschlossene Bestpreisauktion, da die Gebote zu einem bestimmten Zeitpunkt an den Auktionator der Gruppe gegeben werden, welcher anschließend die ertragsmaximierende Kombination an Geboten wählt. Andererseits handelt es sich um eine offene Bestpreisauktion, da die Koordination der Agenten auf Basis der offenbarten Präferenzen stattfindet und somit allen Teilnehmern die Preisvorstellungen der anderen Teilnehmer bekannt sind.¹⁰⁵⁶ Gegen eine Kategorisierung als offene Auktionsform spricht jedoch der Umstand, dass die Agenten ihr eigenes Bietverhalten im Rahmen der ihnen zur Entwurfszeit implementierten Pläne nicht an die Kenntnis dieser externen Bewertungen der versteigerten Güter anpassen.

Daher lassen sich Ad-hoc-Auktionen auch nicht dezidiert dem *independent private values model* zuordnen.¹⁰⁵⁷ Einerseits sind die Bewertungen der versteigerten Güter voneinander unabhängig und erfolgen auf Basis der privaten Bewertungen der menschlichen Benutzer der Agenten. Andererseits sind diese Bewertungen nicht privat, sondern allen Teilnehmern der Auktion bekannt. Dieser Umstand hat jedoch keine Auswirkungen auf das Bietverhalten der Agenten.

6.6.6 Relevanz für die prototypische Implementierung

Die Verwendung eines allgemeinen Modells hat für die prototypische Implementierung zwei entscheidende Vorteile:

1. Es reduziert die Komplexität des mathematischen Modells bei der Implementierung des prototypischen Systems.
2. Es schafft breite Anwendungsmöglichkeiten der entwickelten Technologie jenseits logistik-spezifischer Auktionen.¹⁰⁵⁸

Die Durchführung spontaner Auktionen unter der Eigenregie der anbietenden Agenten innerhalb der gebildeten Gruppen reduziert nicht nur den Koordinationsaufwand, sondern

¹⁰⁵⁴ Die Wahl der Zielfunktion begründet sich folgendermaßen: Die Konzeption und Implementierung der Ad-hoc-Auktion soll die Anforderungsanalyse im Bereich innovativer Online-Frachtenbörsen unterstützen. Es erscheint plausibel, dass potentielle Nutzer einer Online-Frachtenbörse das Ziel der Ertragsmaximierung leichter verstehen als das Ziel allokativer Effizienz.

¹⁰⁵⁵ Vgl. Kapitel 5.3.3.4, S. 103.

¹⁰⁵⁶ Vgl. Kapitel 5.3.3.4, S. 103.

¹⁰⁵⁷ Vgl. Kapitel 5.3.1, S. 96.

¹⁰⁵⁸ Mögliche weitere Anwendungsgebiete der entwickelten Technologie werden in Kapitel 10.3 diskutiert. Vgl. Kapitel 10.3, S. 250.

auch die benötigte Rechenleistung. Der Grund hierfür liegt in der reduzierten Bewertungskomplexität sowie der geminderten strategischen Komplexität.¹⁰⁵⁹ Da alle Agenten kompatible Interessen haben, werden keine unnötigen Nachrichten oder indifferente Präferenzen ausgetauscht – kein Agent erhält Nachrichten, die für ihn irrelevant sind, und jeder Agent hat für mindestens eines der versteigerten Güter eine Präferenz. Hierdurch werden die Kommunikationskomplexität gegenüber einer Auktion mit zentralem Auktionator sowie generell die Zeitkomplexität des Allokationsproblems reduziert.¹⁰⁶⁰

6.7 Zusammenführung der Erkenntnisse

Die wichtigsten Eigenschaften der zu implementierenden Online-Frachtenbörse sind:

- Die Agenten müssen sich selbstständig über das Internet finden und Kontakt herstellen können.
- Sie müssen ihre Präferenzen bzgl. des Angebots von und der Nachfrage nach Transportdienstleistungen in einem standardisierten Format austauschen, um zu ermitteln, mit welchen anderen Agenten sie in Verhandlungen treten möchten.
- Zur Notation dieser Präferenzen ist es notwendig, dass die Transporte auf Basis desselben, abstrakten Verkehrsnetzes beschrieben werden.
- Um den Aufwand bzgl. der notwendigen Koordination und Rechenleistung zu minimieren, finden Auktionen in Gruppen von Agenten statt, die sich bereits vor Beginn der Auktion darüber verständigt haben, dass sie über miteinander kompatible Handelsinteressen verfügen.
- Damit diese, in Bezug auf ihre Präferenzen homogenen Gruppen zustande kommen, muss ein im Ergebnis stabiler Gruppenbildungsprozess stattfinden.
- Die Auktionen werden von einem Agenten innerhalb der Gruppe anhand des zuvor festgelegten Auktionsmodells als Auktionator durchgeführt.
- Die Ergebnisse jeder Auktion werden von allen Beteiligten signiert, bevor die Gruppe in einen finalen Zustand versetzt und hierdurch aufgelöst wird.

Mit Abschluss jeder Ad-hoc-Auktion beginnt dieser „Lebenszyklus“ der Agenten von neuem. Alle Agenten können mehrere dieser Zyklen parallel durchlaufen, also Teilnehmer in verschiedenen Gruppen und Auktionen zu verschiedenen Gütern sein, während sie zusätzlich Kontakthanfragen stellen und beantworten.

Im vorliegenden Kapitel wurden somit alle grundlegend notwendigen Aspekte einer agentenorientierten Online-Frachtenbörse konzipiert. Es wurde deutlich gemacht, wie das hier beschriebene, individuelle Agentenverhalten zu dem erwünschten Gruppenverhalten führt.

¹⁰⁵⁹ Vgl. Kapitel 5.3.3.4, S. 103.

¹⁰⁶⁰ Vgl. Kapitel 5.3.3.4, S. 103.

7 Implementierung einer agentenorientierten Online-Frachtenbörse

7.1 Ziele der prototypischen Implementierung

Die prototypische Implementierung der beschriebenen Aspekte in eine funktionsfähige Software verfolgt das Ziel, die beschriebenen Konzepte auf ihre Realisierbarkeit zu prüfen. Die Konzepte sollen in einem erlebbaren System derart umgesetzt werden, dass Rückschlüsse auf die Anwendbarkeit der konzipierten Agenten in der realen Welt möglich werden. Zu diesem Zweck muss die prototypische Implementierung jedoch nicht nur funktional sein und alle beschriebenen Abläufe implementieren, sondern auch eine grafische Benutzeroberfläche enthalten, welche demonstriert, in welcher Weise die Agenten durch menschliche Benutzer gesteuert werden können.

Das folgende Kapitel beschreibt diese prototypische Implementierung eines Multi-Agenten-Systems. Die Implementierungsarbeiten umfassen die Implementierung einer integrierten Entwicklungsumgebung¹⁰⁶¹ einerseits sowie der „eigentlichen“ Agenten-Software¹⁰⁶² andererseits.

7.2 Implementierungstechniken

7.2.1 In Betracht gezogene Implementierungstechniken

Im Folgenden werden Techniken ausgewählt, welche zur prototypischen Implementierung der Software-Agenten eingesetzt werden sollen.¹⁰⁶³ Diese Implementierungstechniken sollen einigen strengen, noch zu identifizierenden Kriterien genügen, um die Implementierung des Softwareprototyps im Rahmen dieser Forschungsarbeit zu ermöglichen.

Die ausgewählten Implementierungstechniken sind:

- Eine Agentenprogrammiersprache, welche sowohl die Implementierung der Agenten als auch einer integrierten Entwicklungsumgebung ermöglicht,
- ein einheitliches Datenformat zum Informationsaustausch zwischen einzelnen Bestandteilen der Software sowie

¹⁰⁶¹ Vgl. Kapitel 7.3, S. 190.

¹⁰⁶² Vgl. Kapitel 7.4, S. 215.

¹⁰⁶³ Die in den folgenden Kapiteln dargestellten Überlegungen und Argumentationen zu Implementierungstechniken wurden vom Autor im Kern bereits an anderer Stelle publiziert. Vgl. FÖHRING/BRUNS (2011), S. 10 ff.

- ein Datenbanksystem zur Speicherung von Daten wie bspw. Verkehrsnetzen, Agentenkonfigurationen und Handelspräferenzen,
- einen Webserver zur Bereitstellung einer internetbasierten Benutzeroberfläche sowie
- eine Versionsverwaltung zur Vorhaltung des Quelltextes der Software während und nach Abschluss der Implementierungsarbeiten.

Die zentrale Rolle nimmt hierbei die Auswahl der Agentenprogrammiersprache ein. Eine weniger endgültige Entscheidung stellt die Auswahl des verwendeten Datenbanksystems und Webserver sowie der verwendeten Versionsverwaltung dar, da die Auswahl dieser Technologien zu einem späteren Zeitpunkt mit vertretbarem Aufwand modifiziert werden kann.¹⁰⁶⁴

Ebenfalls ausgewählt werden folgende Implementierungstechniken:¹⁰⁶⁵

- Browsertechnologien sowie
- die zur Entwicklung eingesetzte Betriebssystem/Quelltexteditor-Kombination.

Mit dem Begriff „Browsertechnologien“ werden im Folgenden jene Auszeichnungs- und Skriptsprachen zusammengefasst, die lokal auf dem Rechner des Benutzers ihren Einsatz finden, um die Webseite darzustellen.

Zu den Browsertechnologien HTML, CSS und JavaScript gibt es im Internet keine wirtschaftlich einsetzbaren Alternativen:

- Anstatt des HTML-Formats könnte theoretisch auch ein beliebiges XML-Format verwendet werden, welches in einem zweiten Schritt per CSS visuell aufbereitet werden müsste.¹⁰⁶⁶
- Anstatt CSS könnten Attribute nach dem Standard „HTML 4“ verwendet werden, um die Farbe und Form einzelner Elemente zu beeinflussen, allerdings würde diese Vorgehensweise wesentlich größere HTML-Dateien verursachen und die Flexibilität stark einschränken.¹⁰⁶⁷
- Zu JavaScript gäbe es zumindest theoretisch die Alternative VBScript einzusetzen, die allerdings nur von Browsern des Herstellers Microsoft interpretiert und von Microsoft auch nicht mehr weiterentwickelt wird und so den Anwenderkreis der Benutzeroberfläche originär stark einschränken würde.¹⁰⁶⁸

¹⁰⁶⁴ Der Grund hierfür ist, dass die Webserver- und Datenbanktechnologien lediglich unterstützende Technologien zum Betrieb des Softwareprototyps sind. Die Versionsverwaltung ist ebenfalls lediglich eine unterstützende Technologie zur Entwicklung und Verwaltung des Quelltextes des Softwareprototyps.

¹⁰⁶⁵ Die Auswahl dieser Implementierungstechniken geschieht indirekt über die Auswahl der zuvor genannten Implementierungstechniken.

¹⁰⁶⁶ Vgl. ROTHFUSS/RIED (2003), S. 82 f.

¹⁰⁶⁷ Vgl. Kapitel 7.2.3.7, S. 181.

¹⁰⁶⁸ Vgl. LIPPERT (2004), o.S.

Bei der Auswahl der Betriebssystem/Quelltexteditor-Kombination kann es theoretisch, je nach Programmiersprache, Datenbanksystem, Webserver und Versionsverwaltung, zu Einschränkungen kommen, wenn nur einzelne Betriebssysteme bzw. Betriebssystemfamilien (bspw. Microsoft Windows, Linux oder Apple Mac OS X) von der gewählten Technologiekombination unterstützt werden. Analog wird es durch die Auswahl entsprechender Technologien möglich, bei der Implementierung und Wartung des Softwareprototyps nur geringe Anforderungen an die verfügbare technische Ausstattung zu stellen. Dies ist relevant, da der Softwareprototyp der weiteren Anwendung und Forschung möglichst niedrige Einstiegshürden bieten soll.

Wie gezeigt wurde, existiert bei den Browsertechnologien und der Betriebssystem/Quelltexteditor-Kombination keine direkte Auswahlproblematik. Die Auswahl dieser Implementierungstechniken ergibt sich indirekt aus der Auswahl der direkt ausgewählten Implementierungstechniken. Ihre Auflistung dient lediglich einer vollständigeren Darstellung der eingesetzten Implementierungstechniken.

7.2.2 Kriterien zur Bewertung der Implementierungstechniken

Da zur prototypischen Implementierung der Software-Agenten lediglich der finanzielle und personelle Rahmen einer Forschungsarbeit zur Verfügung steht, sollen bei der Implementierungsarbeit Technologien zum Einsatz kommen, welche eine schnelle, praxis- und problemorientierte Implementierung geplanter Funktionalitäten unterstützen.

Es werden teilweise Technologien identifiziert, welche für die prototypische Implementierung im Rahmen dieser Forschungsarbeit als ungeeignet erachtet werden, jedoch im Falle einer professionellen Implementierung für einen späteren Produktiveinsatz¹⁰⁶⁹ besser geeignet wären als die gewählten Technologien. In diesen Fällen wird explizit darauf hingewiesen.

Als wichtige erachtete Eigenschaften der zu evaluierenden Technologien sind daher:

- geringe (monetäre) Kosten,¹⁰⁷⁰
- geringe Entwicklungszeit¹⁰⁷¹ sowie

¹⁰⁶⁹ Der Unterschied zwischen einer prototypisch implementierten Software und einer Software für den Produktiveinsatz ist nicht trennscharf definiert. Für die Zwecke dieser Forschungsarbeit liegt er in einer höheren Anzahl an heterogenen Benutzern, welche die Software ohne Anleitung mit einer eigenen, potentiell mit monetärer Gewinnabsicht verknüpften Erwartungshaltung nutzen.

¹⁰⁷⁰ Alle im Folgenden beschriebenen Technologien sind „Open Source“. Sie sind somit quelloffen verfügbar und entgeltfrei nutzbar. Die Lizenzen der hier beschriebenen Technologien erlauben zudem explizit die kommerzielle Nutzung.

¹⁰⁷¹ Eine geringe Entwicklungszeit wird immer dann begünstigt, wenn zur Implementierung geplanter Funktionalitäten möglichst wenig Vorarbeit, bspw. in Form von Installation und Konfiguration zusätzlicher Komponenten, nötig ist. Neue Ideen sollten im Idealfall direkt implementiert und zusätzliche Funktionen so schnell auf ihre Praxisrelevanz getestet werden können.

- hohe Flexibilität¹⁰⁷² bei gleichzeitiger Bewährtheit.¹⁰⁷³

Als weniger wichtig werden dagegen folgende Eigenschaften erachtet:

- hohe Ausführungsgeschwindigkeit des Softwareprototyps¹⁰⁷⁴ sowie
- hohe Sicherheit¹⁰⁷⁵.

Es wären weitere Kriterien zur Bewertung der vorgestellten Implementierungstechniken denkbar, wie bspw. die Qualifikation, Verfügbarkeit und Kosten des zur Wartung und zum Betrieb der Software nötigen Fachpersonals.¹⁰⁷⁶ Auf Kriterien dieser Art wurde verzichtet, da sich diese Forschungsarbeit nicht mit der Auswahl von Technologien zur Gründung einer tatsächlichen Online-Frachtenbörse beschäftigt, sondern mit der erstmaligen Exploration der Problemdomäne „agentenorientierte Online-Frachtenbörsen“ und der Implementierung eines Softwareprototyps.

Die aufgeführten Eigenschaften bilden die Kriterien zur Auswahl der Implementierungstechniken des Softwareprototyps. Vor dem Hintergrund dieser Kriterien wird entschieden, dass lediglich Technologien in Betracht gezogen werden, welche als „Open-Source-Software“ zur Verfügung stehen. Die prototypische Implementierung des in dieser Forschungsarbeit behandelten Systems zielt schon der Bezeichnung wegen nicht auf ein finales, für den Produktiveinsatz vorgesehenes Softwaresystem ab. Es ist vielmehr Anliegen dieser Forschungsarbeit, die weitere Diskussion und Forschung im Bereich von Online-Frachtenbörsen auf Basis von Multi-Agenten-Systemen voranzubringen.

Die Verwendung von kostenlosen und quelloffenen Technologien zur Erstellung des Softwareprototyps hat zur Folge, dass die hier dargestellten technologischen Entwicklungen ebenfalls als Open-Source-Software veröffentlicht werden können, ohne Lizenzierungsprobleme durch die teilweise Verwendung kommerzieller Software befürchten zu müssen. Zudem ermöglicht dieser Schritt interessierten Wissenschaftlern weltweit und unabhängig

¹⁰⁷² Mit Flexibilität ist vor allem gemeint, dass keine strengen Paradigmen existieren sollen, die den Entwickler bei der Verwendung der Technologie einschränken. Der Entwickler sollte möglichst frei in seinen Entscheidungen bzgl. der eingesetzten Entwicklungsumgebung, Betriebssysteme und Datenformate sein. Ferner sollten sich diese Entscheidungen im Laufe der explorativen Phase der Entwicklung mit möglichst wenig Aufwand abändern lassen.

¹⁰⁷³ Mit Bewährtheit ist vor allem gemeint, dass keine Programmiersprache gewählt werden soll, die den zuvor genannten Kriterien zwar bei theoretischer Betrachtung genügt, jedoch in der praktischen Anwendung unter einem Mangel zur Verfügung stehender Ressourcen, wie bspw. Lehrbüchern, unterstützenden Internetforen und sorgfältig getesteten Softwarebibliotheken, leidet.

¹⁰⁷⁴ Für einen Softwareprototyp ist die Ausführungsgeschwindigkeit zunächst sekundär, da wenige Benutzer den Softwareprototyp gleichzeitig nutzen werden, da er lediglich der weiteren Anforderungsanalyse dient und nicht produktiv eingesetzt werden soll.

¹⁰⁷⁵ Fragen bzgl. der Systemsicherheit, Typsicherheit oder Threadsicherheit fließen nicht in die Bewertung ein, da der Softwareprototyp nicht für den Produktiveinsatz implementiert wird.

¹⁰⁷⁶ Denkbar wären Überlegungen wie bspw., dass das Arbeitskräfteangebot bei Open-Source-Technologien (aufgrund des offenen Zugangs) tendenziell größer sein dürfte als bei geschlossenen Technologien oder dass Entwickler, die sich geschlossener Technologien bedienen, höher qualifiziert sein könnten (aufgrund einer tendenziell strukturierteren Ausbildung).

von den ihnen zur Verfügung stehenden Mitteln die hier gezeigte Forschung nachzuvollziehen, zu reproduzieren und zu erweitern. Die Einstiegshürden für weitere Forschungsbeiträge zur behandelten Thematik werden so weit gesenkt, dass Wohlstandsunterschiede, bspw. in Bezug auf den Zugang zu kommerziell verfügbarer Software oder proprietären Daten, zwischen verschiedenen Institutionen keine entscheidende Rolle spielen und auch Wissenschaftler ohne institutionelle Beziehungen in die Lage versetzt werden, eigene Forschungsbeiträge zu erstellen.

In Fällen, in denen eine kommerzielle Alternative Vorteile, wie bspw. eine gesteigerte Genauigkeit oder erhöhte Ausführungsgeschwindigkeit, gegenüber den in der vorliegenden Implementierung verwendeten offenen Technologien, Formaten oder Daten bietet, wird an entsprechender Stelle explizit darauf hingewiesen.

7.2.3 Beschreibung der Implementierungstechniken

7.2.3.1 Agentenprogrammiersprache

Bedeutung für die Implementierungsarbeit

Es muss eine Programmiersprache als Agentenprogrammiersprache gewählt werden, die den zuvor genannten Kriterien besonders streng genügt, da ihre Auswahl bereits Auswirkungen auf die Eigenschaften des späteren Softwareprototyps haben kann.¹⁰⁷⁷

Alle beschriebenen Programmiersprachen sind kostenlos verfügbar und verfügen über eine breite Anhängerschaft unter professionellen Entwicklern, wodurch die Kriterien der geringen Kosten und der Verfügbarkeit von vorgefertigten Lösungsschemata bei allen beschriebenen Programmiersprachen erfüllt sind.

Die Auswahl der Programmiersprache ist für die Implementierung des Softwareprototyps deshalb von zentraler Bedeutung, da sie den Rahmen vorgibt, innerhalb dessen sich die Möglichkeiten der zu entwickelnden Software ergeben. Grundsätzlich ist festzuhalten, dass mit allen zur Auswahl gestellten Sprachen das Ziel, ein Multi-Agenten-System zu implementieren, erreicht werden kann. Die vorgestellten Programmiersprachen unterscheiden sich jedoch hinsichtlich ihrer Erfüllung der zuvor definierten Kriterien für in Betracht gezogene Implementierungstechniken. Da keine der vorgestellten Programmiersprachen alle der Kriterien erfüllt, wird im Folgenden ein Überblick gegeben, um anschließend eine begründete Auswahl des *best fit* treffen zu können.

Prinzipiell stellt die Konzeption jeder Programmiersprache eine Abwägung der konfliktären Ziele Funktionalität, Ausführungsgeschwindigkeit und Portabilität dar.¹⁰⁷⁸

¹⁰⁷⁷ So könnte die Auswahl einer plattformabhängigen Programmiersprache dazu führen, dass der Softwareprototyp bspw. nur unter Windows-Betriebssystemen mit vertretbarem Aufwand weiterentwickeln lässt.

¹⁰⁷⁸ Vgl. NIEMEYER/KNUDSEN (2005), S. 8.

Zur Auswahl stehen kompilierbare Programmiersprachen und Skriptsprachen.

- **Kompilierbare Programmiersprachen:** Der Quelltext wird zu einem festen Zeitpunkt kompiliert, d.h. in eine Maschinensprache übersetzt.¹⁰⁷⁹ Maschinensprachen sind – vereinfacht dargestellt – übersetzte Varianten des Quelltextes, welche nicht für den Menschen, sondern für den Prozessor lesbar sind und von diesem ausgeführt werden können. Deshalb besitzen Programme, welche in kompilierbaren Programmiersprachen geschrieben sind, regelmäßig eine höhere Ausführungsgeschwindigkeit als solche, welche nicht in kompilierbaren Programmiersprachen geschrieben wurden.¹⁰⁸⁰ Während des Kompilierungsprozesses finden Sicherheitsüberprüfungen statt, so dass die resultierenden Applikationen¹⁰⁸¹ regelmäßig sicherer in der Ausführung sind als analog in einer Skriptsprache geschriebene Applikationen. Aus demselben Grund sind kompilierbare Programmiersprachen nicht dynamisch: ihr Quelltext lässt sich nach dem Kompilierungsprozess zur Laufzeit nicht mehr abändern.¹⁰⁸²
- **Skriptsprachen:** Der Quelltext von Skriptsprachen wird nicht zu einem festen Zeitpunkt kompiliert, sondern zur Laufzeit der Applikation von einem Interpreter übersetzt und ausgeführt.¹⁰⁸³ Aufgrund ihrer Flexibilität und Fehlertoleranz sind sie mächtige Werkzeuge zur schnellen Entwicklung von Softwareprototypen.¹⁰⁸⁴

Den Betrachtungen in Kapitel 5.2.9 folgend werden lediglich objektorientierte Programmiersprachen betrachtet.¹⁰⁸⁵ Es werden jeweils zwei kompilierbare Programmiersprachen (C# und Java) sowie zwei Skriptsprachen (Ruby und Python) beschrieben.

Zusätzlich bieten einige der zur Auswahl stehenden Sprachen die Möglichkeit des Einsatzes von Metaprogrammierung. Metaprogrammierung ermöglicht einem Programm, seine eigene Struktur zu verändern, d.h. sich selbst umzuschreiben, um zur Laufzeit auf veränderte Rahmenbedingungen zu reagieren. Dieses „Umschreiben des eigenen Quelltextes“ bietet das Potential, Agenten zu entwickeln, welche mit sich aktualisierenden Plänen auf ihre Umgebung reagieren können.

Idealerweise sollte die auszuwählende Programmiersprache sich zur Implementierung sowohl einer integrierten Entwicklungsumgebung als auch der Agenten selbst eignen.

Folgende Programmiersprachen stehen zur Auswahl.

¹⁰⁷⁹ Vgl. NIEMEYER/KNUDSEN (2005), S. 8.

¹⁰⁸⁰ Vgl. NIEMEYER/KNUDSEN (2005), S. 8.

¹⁰⁸¹ Der Begriff „Applikation“ wird im Folgenden als Synonym für eine Anwendungssoftware verwendet.

¹⁰⁸² Vgl. NIEMEYER/KNUDSEN (2005), S. 8.

¹⁰⁸³ Vgl. NIEMEYER/KNUDSEN (2005), S. 8.

¹⁰⁸⁴ Vgl. NIEMEYER/KNUDSEN (2005), S. 8.

¹⁰⁸⁵ Vgl. Kapitel 5.2.9, S. 93.

C#

C# ist eine von dem Unternehmen „Microsoft“ entwickelte Programmiersprache zur strukturierten, komponentenbasierten und objektorientierten Implementierung großer Applikationen.¹⁰⁸⁶ Das Ziel von C# ist es, eine einfache, sichere und hoch performante Programmiersprache zu sein.¹⁰⁸⁷ Die Konzeption und Syntax von C# lehnt sich an Programmiersprachen wie C++ und Java an.¹⁰⁸⁸

C# ist über das Internet kostenlos verfügbar.¹⁰⁸⁹

Mit C# lassen sich Applikationen kompilieren, allerdings wird hierbei kein direkt vom Prozessor ausführbarer Maschinencode erzeugt.¹⁰⁹⁰ Vielmehr übersetzt der Compiler den in C# verfassten Quelltext zunächst in eine intermediäre Maschinsprache namens Microsoft Intermediate Language (MSIL) und speichert diese Informationen zusammen mit anderen zur Ausführung notwendigen Daten in eine ausführbare Datei.¹⁰⁹¹ Wird diese Datei ausgeführt, liest der .NET-JIT-Compiler den MSIL-Code, übersetzt ihn in die native Maschinsprache des Prozessors und erzeugt hieraus zur Laufzeit eine ausführbare Applikation im Arbeitsspeicher.¹⁰⁹² C# unterstützt als Sprache keine komplexe Metaprogrammierung.

C# ist als Programmiersprache Teil der von Microsoft entwickelten .NET-Plattform zur Entwicklung und Ausführung von Windows- und Webapplikationen sowie Webservices.¹⁰⁹³ Als Teil dieser Software-Plattform hat C# Zugriff auf die umfangreichen Klassenbibliotheken von .NET.¹⁰⁹⁴

¹⁰⁸⁶ Vgl. LIBERTY/MACDONALD (2009), S. 5.

¹⁰⁸⁷ Vgl. LIBERTY/MACDONALD (2009), S. 5.

¹⁰⁸⁸ Vgl. NASH (2010), S. 17, LIBERTY/MACDONALD (2009), S. 2 u. 5, sowie NIEMEYER/KNUDSEN (2005), S. 4.

¹⁰⁸⁹ C# wird als Teil von Microsofts Visual Studio Express vertrieben. Weitere Informationen sind im Internet verfügbar. Vgl. MICROSOFT (2017).

¹⁰⁹⁰ Vgl. SOLIS (2012), S. 7 ff., sowie LIBERTY/MACDONALD (2009), S. 10.

¹⁰⁹¹ Vgl. NASH (2010), S. 10 ff., LIBERTY/MACDONALD (2009), S. 10, sowie SOLIS (2012), S. 7 ff.

¹⁰⁹² Vgl. SOLIS (2012), S. 7 ff., NASH (2010), S. 10 ff., sowie LIBERTY/MACDONALD (2009), S. 10.

¹⁰⁹³ Vgl. LIBERTY/MACDONALD (2009), S. 5 u. 17.

Die Plattform unterstützt mehrere Sprachen und steht im vollen Umfang nur für Windows zur Verfügung, wenngleich Open-Source-Compiler für C# auch für unixoide Systeme existieren, bspw. das Mono-Projekt und #develop. Vgl. NASH (2010), S. 15, LIBERTY/MACDONALD (2009), S. 17, sowie SOLIS (2012), S. 5. Die CLR-Komponente ist neben der Code-Ausführung auch für das Speichermanagement, die Sicherheitsverifikation und die automatische Speicherbereinigung zuständig. Vgl. SOLIS (2012), S. 3 f. u. 6.

¹⁰⁹⁴ Vgl. LIBERTY/MACDONALD (2009), S. 17.

Java

Java ist eine plattformunabhängige, schnelle und sichere Programmiersprache.¹⁰⁹⁵ Das Ziel von Java ist, eine einfache Programmiersprache zu sein, auf deren Grundlage komplexere Konstrukte entwickelt werden können.¹⁰⁹⁶ Die Syntax von Java lehnt sich an die Programmiersprache C an.¹⁰⁹⁷ Java ist sowohl im Design als auch in der Implementierung sicher.¹⁰⁹⁸

Java ist über das Internet kostenlos verfügbar.¹⁰⁹⁹

Mit Java lassen sich Applikationen kompilieren, allerdings wird hierbei kein nativ von der Prozessorarchitektur ausführbarer Maschinencode erzeugt.¹¹⁰⁰ Der Java-Compiler erzeugt aus dem Quelltext zuerst einen Bytecode, der anschließend von einer Java Virtual Machine (JVM) ausgeführt werden kann.¹¹⁰¹ Die Ausführungsgeschwindigkeit dieses Bytecodes ist aufgrund eines JIT-Compilers ähnlich zur Ausführungsgeschwindigkeit nativ kompilierter Applikationen.¹¹⁰²

Java eignet sich zur Entwicklung komplexer Webapplikationen, da Java-Applikationen regelmäßig eine höhere Ausführungsgeschwindigkeit haben als Applikationen, die in Skriptsprachen geschrieben sind.¹¹⁰³ Durch Javas Plattformunabhängigkeit können Webapplikationen schnell und einfach auf andere Hardwarekonfigurationen migriert werden. Zudem existieren weitere Technologien wie Java Server Pages, Java Server Faces und XML, die das Entwickeln komplexer Webapplikationen zusätzlich vereinfachen und unterstützen.¹¹⁰⁴

Java unterstützt als Sprache keine komplexe Metaprogrammierung.¹¹⁰⁵

Ruby

Ruby ist eine objektorientierte Programmiersprache, welche Funktionalitäten zur Metaprogrammierung bietet. Da diese Metaprogrammierung in Ruby selbst zur Laufzeit stattfindet, eignet es sich zur Programmierung eigener domänenspezifischer Sprachen

¹⁰⁹⁵ Vgl. NIEMEYER/KNUDSEN (2005), S. 1.

¹⁰⁹⁶ Vgl. NIEMEYER/KNUDSEN (2005), S. 10.

¹⁰⁹⁷ Vgl. NIEMEYER/KNUDSEN (2005), S. 83.

¹⁰⁹⁸ Vgl. NIEMEYER/KNUDSEN (2005), S. 9 ff. u. 15 ff.

¹⁰⁹⁹ Weitere Informationen sind im Internet verfügbar. Vgl. ORACLE (2017).

¹¹⁰⁰ Vgl. NIEMEYER/KNUDSEN (2005), S. 4 f.

¹¹⁰¹ Vgl. NIEMEYER/KNUDSEN (2005), S. 4 f.

¹¹⁰² Beispiele für JIT-Compiler sind die JVM von Java und die CLR von .NET. Vgl. Kapitel 7.2.3.1, S. 161.

¹¹⁰³ Vgl. NIEMEYER/KNUDSEN (2005), S. 8.

¹¹⁰⁴ Vgl. NIEMEYER/KNUDSEN (2005), S. 502 f.

¹¹⁰⁵ Vgl. TATE (2005), S. 11.

(DSL).¹¹⁰⁶ Ruby lehnt sich an Konzepte der Programmiersprachen Smalltalk und Perl an, wurde aber so konzipiert, dass es Entwicklern, die mit C und Java vertraut sind, leicht fällt, Ruby zu lernen.¹¹⁰⁷

Ruby ist über das Internet kostenlos verfügbar.¹¹⁰⁸

Ruby ist eine Skriptsprache, die während der Laufzeit interpretiert wird und, im Gegensatz zu C# und Java, zur Ausführung nicht kompiliert werden muss. Allein dieser Faktor beschleunigt die Entwicklungszeit¹¹⁰⁹ zu Lasten der Ausführungsgeschwindigkeit.¹¹¹⁰

Ruby ist vollständig objektorientiert – jeder Wert ist ein Objekt. Dennoch eignet sich Ruby auch für prozedurale und funktionale Programmierstile.¹¹¹¹

Die Programmiersprache Ruby eignet sich besonders gut zur schnellen Implementierung von Prototypen, da sie als Sprache konzipiert ist, die es dem Programmierer erleichtern soll, sich auf sein Problem zu konzentrieren, ohne dabei permanent auf die Besonderheiten der Programmiersprache achten zu müssen.¹¹¹²

Python

Python ist eine laufzeitinterpretierte Programmiersprache, welche verschiedene Paradigmen moderner Programmierung unterstützt, bspw. objektorientierte, imperative und funktionale Programmierung.¹¹¹³

Python ist über das Internet kostenlos verfügbar.¹¹¹⁴

Python wurde in den 1990ern Jahren von Guido van Rossum entwickelt.¹¹¹⁵

Die Entwickler der Sprache folgen der Philosophie, dass nicht alle Spracheigenschaften im Kern der Sprache implementiert sein müssen, sondern dass die Sprache so modular

¹¹⁰⁶ Eine DSL ist eine domänenspezifische Sprache für ein spezielles Problemfeld. Vgl. GAMBLE et al. (2010), S. 4.

¹¹⁰⁷ Vgl. FLANAGAN/MATSUMOTO (2008), S. 2.

¹¹⁰⁸ Weitere Informationen sind im Internet verfügbar. Vgl. RUBY (2017).

¹¹⁰⁹ Der Grund hierfür ist, dass der Quelltext nicht nach jeder Modifikation (bspw. der Berichtigung eines Tippfehlers im Quelltext) neu kompiliert werden muss, was je nach Ausstattung des Rechners, der Komplexität des Projekts und der Programmiersprache schnell Minuten in Anspruch nehmen kann.

¹¹¹⁰ Vgl. CARLSON/RICHARDSON (2006), S. 817.

¹¹¹¹ Selbst primitive Typen, die booleschen Werte „true“ und „false“ sowie das die Abwesenheit eines Wertes repräsentierende „nil“ sind in Ruby Objekte. Vgl. FLANAGAN/MATSUMOTO (2008), S. 2.

¹¹¹² Der Schöpfer von Ruby sagt, er habe stets sich selbst als Zielgruppe vor Augen gehabt: einen einfachen Programmierer, der eine Sprache nutzen möchte, die ihm die Arbeit erleichtert („Ruby is designed to make programmers happy.“). Vgl. FLANAGAN/MATSUMOTO (2008), S. 2.

¹¹¹³ Vgl. MARTELLI (2006), S. 3 ff.

¹¹¹⁴ Weitere Informationen sind im Internet verfügbar. Vgl. PYTHON (2017).

¹¹¹⁵ Vgl. MARTELLI (2006), S. 3.

angelegt sein sollte, dass fehlende Funktionalitäten von Programmierern nachträglich implementiert werden können.¹¹¹⁶ Python nutzt dynamische Typisierung und ist durch C erweiterbar.¹¹¹⁷

LANGTANGEN beschreibt die Vorteile von Python zur Durchführung wissenschaftlicher Berechnungen.¹¹¹⁸ Wie auch Ruby unterstützt Python das Paradigma der aspektorientierten Programmierung durch Meta-Programmierung und ermöglicht die Erzeugung von neuen Programmcodes zur Laufzeit.¹¹¹⁹

7.2.3.2 Datenformat

Bedeutung für die Implementierungsarbeit

Eine einheitliche Formatierung der Daten des Softwareprototyps ist notwendig, damit die Agenten diese mit Hilfe agenten-spezifischer Sprachen austauschen können. Wenngleich diese Formatierung zu einem späteren Zeitpunkt noch angepasst werden kann, stellt ihre Auswahl potentiell eine grundlegende Designentscheidung dar. Hierdurch lohnt sich die Analyse einiger Alternativen zur Reflexion ihrer Erfüllung der angelegten Kriterien.

Folgende Datenformate stehen zur Auswahl.

XML

Die „Extensible Markup Language“ (XML) ist eine Meta-Auszeichnungssprache zur textlichen Darstellung hierarchisch strukturierter Daten.¹¹²⁰ XML definiert eine generische Syntax, um Daten mit menschenlesbaren Annotationen zu versehen und bietet ein standardisiertes Format, welches flexibel genug ist, um Daten aus verschiedensten Domänen zu beschreiben, wie bspw. Webseiten, Vektorgraphiken, Produktkataloge oder Objektserialisierungen.¹¹²¹

Sie ist eine Meta-Auszeichnungssprache für Textdokumente.¹¹²² Im Gegensatz zu einer Auszeichnungssprache wie bspw. HTML definiert die XML-Spezifikation nicht explizit die verwendbaren Auszeichnungselemente, sondern nur deren Gestaltung und Format.¹¹²³

¹¹¹⁶ Vgl. MARTELLI (2006), S. 4 f.

¹¹¹⁷ Vgl. MARTELLI (2006), S. 3 u. 614 ff.

¹¹¹⁸ Im Englischen „computational science“. Vgl. LANGTANGEN (2009), S. 20 f. u. 131 ff.

¹¹¹⁹ Vgl. LANGTANGEN (2009), S. 13 f., sowie MARTELLI (2006), S. 101 ff.

¹¹²⁰ XML ist ein Subset der Dokumentensprache SGML. Vgl. BRAY et al. (2006), S. 1. Weitere Informationen sind im Internet verfügbar. Vgl. W3 (2017).

¹¹²¹ Vgl. HAROLD/MEANS (2009), S. 3.

¹¹²² Vgl. HAROLD/MEANS (2009), S. 4.

¹¹²³ Vgl. HAROLD/MEANS (2009), S. 4.

Hierdurch ist XML flexibel einsetzbar und kann auf viele verschiedene Problemdomänen angewendet werden.¹¹²⁴ Die XML-Spezifikation definiert lediglich eine Grammatik, der alle XML-Dokumente folgen müssen.¹¹²⁵ Dokumente, welche die Kriterien der XML-Spezifikation erfüllen, werden als „wohlgeformt“ bezeichnet.¹¹²⁶

Aus Gründen der Interoperabilität einigen sich Anwender von XML häufig auf die Benutzung einer begrenzten Menge an Elementen.¹¹²⁷ Diese Menge wird als XML-Applikation bezeichnet.¹¹²⁸

Eine XML-Applikation entspricht regelmäßig einer Problemdomäne, wie bspw. Vektorgraphiken.¹¹²⁹ Daten werden in XML-Dokumenten in Form von Zeichenketten eingebettet.¹¹³⁰ Die Daten werden umgeben von Auszeichnern, welche die Art und Verwendung der Daten beschreiben. Auf diese Weise können XML-Dokumente Elemente beschreiben, die bspw. als Kreis oder Rechteck interpretiert werden sollen.¹¹³¹ In der Regel sagt ein XML-Dokument jedoch nichts darüber aus, wie die in ihm enthaltenen Daten repräsentiert werden sollen.¹¹³² Die das XML-Dokument ladende Software muss wissen, dass das XML-Dokument bspw. eine Vektorgraphik sein soll.

Trotz seiner Flexibilität ist XML nur eine Auszeichnungssprache.¹¹³³ Die Popularität des Datenformats XML hat in der Vergangenheit zu einigen populären Missverständnissen¹¹³⁴ geführt, wie bspw. dem, dass XML ein Netzwerkprotokoll¹¹³⁵ oder eine Datenbank¹¹³⁶

¹¹²⁴ Vgl. HAROLD/MEANS (2009), S. 4.

¹¹²⁵ Vgl. HAROLD/MEANS (2009), S. 4.

¹¹²⁶ Vgl. HAROLD/MEANS (2009), S. 4.

¹¹²⁷ Vgl. HAROLD/MEANS (2009), S. 4.

¹¹²⁸ Vgl. HAROLD/MEANS (2009), S. 4.

¹¹²⁹ Vgl. HAROLD/MEANS (2009), S. 4.

¹¹³⁰ Vgl. HAROLD/MEANS (2009), S. 4.

¹¹³¹ Vgl. HAROLD/MEANS (2009), S. 4.

¹¹³² Vgl. HAROLD/MEANS (2009), S. 4.

¹¹³³ Vgl. HAROLD/MEANS (2009), S. 5.

¹¹³⁴ Vgl. HAROLD/MEANS (2009), S. 5.

¹¹³⁵ Daten können im XML-Format enkodiert versandt werden, jedoch ist die Transportebene stets ein Protokoll wie bspw. HTTP oder FTP und die Daten müssen von einer Software in das XML-Format enkodiert und versendet werden. Vgl. HAROLD/MEANS (2009), S. 6.

¹¹³⁶ Datenbanken können sowohl Daten im XML-Format enthalten als auch Schnittstellen besitzen, welche Daten aus der Datenbank auf Wunsch im XML-Format zur Verfügung stellen. Datenbanken sind jedoch selbst keine XML-Dokumente. Vgl. HAROLD/MEANS (2009), S. 6.
Dieses Missverständnis wird nicht zuletzt von der Softwarebranche genährt, indem Produkte im Rahmen von Werbemaßnahmen und Veröffentlichungen als „XML Datenbanken“ angepriesen werden. Als Beispiel dient an dieser Stelle die von KAY beschriebene Position des Unternehmens „Software AG“, das im Gegensatz zu anderen Unternehmen eine native XML-Datenbank anbietet.
Vgl. KAY (2003), S. 2 f.

sei. XML ist ein wohldokumentiertes¹¹³⁷, textbasiertes Datenformat.¹¹³⁸ Zum Zeitpunkt seiner Entstehung war XML das portabelste und flexibelste Datenformat für textbasierte Dokumente.¹¹³⁹

Die Auszeichnungselemente, welche in einer bestimmten XML-Applikation valide sind, können in einem Schema dokumentiert werden.¹¹⁴⁰ Mit Hilfe dieses Schemas können Anwender anschließend ihre XML-Dokumente validieren.¹¹⁴¹ Es gibt mehrere Schemasprachen, mit deren Hilfe Anwender Schemata für ihre XML-Dokumente definieren können. Die verbreitetste dieser Sprachen ist *Document Type Definition* (DTD). Eine DTD listet alle erlaubten Auszeichnungselemente auf und beschreibt, wo diese Auszeichnungselemente im Dokument vorkommen dürfen. DTDs erlauben jedoch nicht die Beschreibung von semantischen Anforderungen, wie bspw. „dieses Element ist eine Zahl“ oder „diese Jahreszahl muss zwischen 1980 und 2020 liegen“. Zur Formulierung derartiger Anforderungen dient die *W3C XML Schema Language*. Neben den beiden genannten Schemasprachen existierten viele weitere, deren Anwendungsgebiete sich teilweise überschneiden.¹¹⁴²

Listing 1 zeigt beispielhaft eine Nachricht zwischen zwei Agenten in XML-Notation.

¹¹³⁷ Vgl. BRAY et al. (2006).

¹¹³⁸ Vgl. HAROLD/MEANS (2009), S. 7.

¹¹³⁹ Vgl. HAROLD/MEANS (2009), S. 7.

¹¹⁴⁰ Vgl. HAROLD/MEANS (2009), S. 4.

¹¹⁴¹ Vgl. HAROLD/MEANS (2009), S. 4.

¹¹⁴² Vgl. HAROLD/MEANS (2009), S. 5.

```
<?xml version="1.0"?>
<message>
  <sender>9A01F36D-9ABD-D4D9-80C7-022AF85C82A8</sender>
  <action>Handshake</action>
  <metadata>
    <data>
      <name>host</name>
      <type>String</type>
      <value>127.0.0.1</value>
    </data>
    <data>
      <name>port</name>
      <type>Number</type>
      <value>15001</value>
    </data>
    <data>
      <name>uuid</name>
      <type>String</type>
      <value>9A01F36D-9ABD-D4D9-80C7-022AF85C82A8</value>
    </data>
  </metadata>
</message>
```

Listing 1: Beispielhafte Nachricht in XML-Notation

JSON

Die „javascript object notation“ (JSON) ist ein programmiersprachenunabhängiges, text-basiertes Datenaustauschformat.¹¹⁴³ Es ist ein Subset der Programmiersprache JavaScript und des ECMA-262-Standards.¹¹⁴⁴

JSON besitzt eine Reihe von Eigenschaften, welche seine Eignung für den Datentransfer unterstreichen: Es ist menschen- und maschinenlesbar.¹¹⁴⁵ Es ist kein Dokumentenformat, sondern ein selbstdokumentierendes Format, welches Strukturen und Feldbezeichnungen ebenso festlegt wie die spezifischen Werte dieser Felder.¹¹⁴⁶ Seine einfache Syntax erlaubt es einfache, effiziente und konsistente Parser zu implementieren.¹¹⁴⁷

¹¹⁴³ Vgl. CROCKFORD (2006), S. 6.

Weitere Informationen sind im Internet verfügbar. Vgl. JSON (2017).

¹¹⁴⁴ Vgl. CROCKFORD (2006), S. 6.

¹¹⁴⁵ Vgl. CROCKFORD (2006), S. 7 u. 48.

¹¹⁴⁶ Vgl. CROCKFORD (2006), S. 7 u. 48.

¹¹⁴⁷ Vgl. CROCKFORD (2006), S. 48.

JSON ist im Gegensatz zu XML versionslos.¹¹⁴⁸ Es eignet sich zur Darstellung der allgemeinsten Datenstrukturen in der Informatik: Datensätze, Listen und Bäume.¹¹⁴⁹ Die einfachen in JSON verwendeten Datentypen sind dieselben wie in vielen populären Programmiersprachen, wodurch eine JSON-Struktur während oder nach ihrer Interpretation nicht verändert werden muss, da die Daten bereits wie in der Programmiersprache selbst strukturiert sind.¹¹⁵⁰

Neben den genannten Vorteilen bietet die Verwendung von JSON einige Nachteile.

JSON unterstützt keine native Separierung der Daten in Namensräume.¹¹⁵¹ Jedes Objekt ist in JSON sein eigener Namensraum, seine Felder sind unabhängig von den Feldern aller anderen Objekte.¹¹⁵² JSON nutzt Kontexte, um Mehrdeutigkeiten zu vermeiden, in derselben Art, wie Programmiersprachen dies tun.¹¹⁵³

Wegen der selbstbeschreibenden Natur des Formats existiert kein allgemeiner Validator für JSON-Strukturen.¹¹⁵⁴ Ein wohlgeformte, valide JSON-Struktur bedeutet nicht, dass die enthaltenen Informationen korrekt formatiert oder relevant sind.¹¹⁵⁵ Jede Applikation ist selbst für die Validierung der eingegebenen Daten verantwortlich.¹¹⁵⁶ Diese Verantwortung kann bei Verwendung von JSON nicht delegiert werden.¹¹⁵⁷ Die verantwortlichen Softwareentwickler haben jedoch die Möglichkeit, ihren JSON-Strukturen Metadaten hinzuzufügen, welche sie zur Validierung und Interpretation nutzen können.¹¹⁵⁸

JSON ist im Gegensatz zu XML nicht erweiterbar.¹¹⁵⁹ Es bietet jedoch die nötige Flexibilität, um jede nicht rekursive Datenstruktur abbilden zu können.¹¹⁶⁰ Zudem lassen sich neue Felder zu bestehenden Datenstrukturen hinzufügen ohne die bestehenden Verarbeitungsprogramme zu beeinträchtigen.¹¹⁶¹

¹¹⁴⁸ Vgl. CROCKFORD (2006), S. 30.

¹¹⁴⁹ Vgl. CROCKFORD (2006), S. 48.

¹¹⁵⁰ JSONs Datentyp „object“ entspricht den Konzepten von „record“, „struct“, „object“, „dictionary“, „hash“ und „assoziativen Arrays“ in anderen Programmiersprachen. Vgl. CROCKFORD (2006), S. 49.

¹¹⁵¹ Vgl. CROCKFORD (2006), S. 50.

¹¹⁵² Vgl. CROCKFORD (2006), S. 51.

¹¹⁵³ Vgl. CROCKFORD (2006), S. 51.

¹¹⁵⁴ Vgl. CROCKFORD (2006), S. 50.

¹¹⁵⁵ Vgl. CROCKFORD (2006), S. 54.

¹¹⁵⁶ Vgl. CROCKFORD (2006), S. 54.

¹¹⁵⁷ Vgl. CROCKFORD (2006), S. 54.

¹¹⁵⁸ Diese Metafelder können der interpretierenden Applikation bspw. mitteilen, dass ein bestimmtes Objekt innerhalb der JSON-Struktur mit einer bestimmten Klasse in der Applikation verknüpft ist oder dass ein bestimmtes Feld nicht nur eine Gleitkommazahl, sondern auch ein Geldbetrag ist. Vgl. CROCKFORD (2006), S. 60 f.

¹¹⁵⁹ Vgl. CROCKFORD (2006), S. 50.

¹¹⁶⁰ Vgl. CROCKFORD (2006), S. 55.

¹¹⁶¹ Vgl. CROCKFORD (2006), S. 55.

JSON bietet die Möglichkeit einer einfachen, allgemeinverständlichen Datenrepräsentation.¹¹⁶² Es eignet sich zur Client-Server-Kommunikation und ermöglicht einen sprachunabhängigen Datenaustausch.¹¹⁶³

Listing 2 zeigt beispielhaft eine Nachricht zwischen zwei Agenten in JSON-Notation.

```
{
  "message": {
    "sender": "9A01F36D-9ABD-D4D9-80C7-022AF85C82A8",
    "action": "Handshake",
    "metadata": {
      "host": "127.0.0.1",
      "port": 15001,
      "uuid": "9A01F36D-9ABD-D4D9-80C7-022AF85C82A8"
    }
  }
}
```

Listing 2: Beispielhafte Nachricht in JSON-Notation

DSL

Eine „Domain Specific Language“ (DSL) ist eine ausführbare Spezifikations- oder Programmiersprache, welche Ausdruckskraft auf eine bestimmte Problemdomäne fokussiert (und üblicherweise beschränkt).¹¹⁶⁴

In der Informatik kann wie in anderen Wissenschaften zwischen generischen und spezifischen Ansätzen der Problemlösung unterschieden werden.¹¹⁶⁵ Ein generischer Ansatz verspricht allgemeine Lösungen zu einer Breite von Problemen, welche jedoch für den spezifischen Einzelfall suboptimal sein können.¹¹⁶⁶ Ein spezifischer Ansatz bietet dagegen eine bessere Lösung für eine kleinere Menge an Problemen.¹¹⁶⁷ In der Informatik stellt die Dichotomie von allgemein verwendbaren Programmiersprachen und domänenspezifischen Sprachen eine Ausprägung dieses Sachverhalts dar.¹¹⁶⁸

Dies ist kein neues Problemfeld: Viele später als allgemein einzustufende Programmiersprachen wurden als spezifische Sprachen zur Lösung bestimmter Probleme entwickelt,

¹¹⁶² Vgl. CROCKFORD (2006), S. 57.

¹¹⁶³ Vgl. CROCKFORD (2006), S. 57.

¹¹⁶⁴ Vgl. KOLOVOS et al. (2006), S. 1, sowie VAN DEURSEN et al. (2000), S. 26.

¹¹⁶⁵ Vgl. VAN DEURSEN et al. (2000), S. 26.

¹¹⁶⁶ Vgl. VAN DEURSEN et al. (2000), S. 26.

¹¹⁶⁷ Vgl. VAN DEURSEN et al. (2000), S. 26.

¹¹⁶⁸ Vgl. VAN DEURSEN et al. (2000), S. 26.

wie bspw. Fortran (kaufmännische Anwendungen) und COBOL (numerische Berechnungen).¹¹⁶⁹ Im Bereich Internetentwicklung ist PHP eine verbreitete Programmiersprache, welche ursprünglich entwickelt wurde, um Zugriffe auf die Homepage ihres Erfinders zu protokollieren.¹¹⁷⁰ Auch die Makrosprache in der Endnutzer-Software „Microsoft Excel“ kann als DSL verstanden werden.¹¹⁷¹

DSLs sind in der Regel klein und umfassen eine beschränkte Menge an Notationen und Abstraktionen.¹¹⁷² Sie sind in der Regel deklarative Sprachen, weshalb sie als ausführbare Spezifikations- oder Programmiersprachen gesehen werden können.¹¹⁷³

Die Verwendung einer DSL bietet einige Vorteile:

- Domänenspezifische Sprachen erlauben die Spezifizierung und Formulierung eines Problems in problemnahen Begriffen mit domänenspezifischen Abstraktionen.¹¹⁷⁴
- Beschreibungen in einer domänenspezifischen Sprache sind präzise, wiederverwendbar und selbstdokumentierend.¹¹⁷⁵
- Domänenspezifische Sprachen enthalten domänenspezifisches Wissen und erhöhen die Produktivität, Verlässlichkeit, Wartbarkeit und Portabilität von Software.¹¹⁷⁶
- Domänenspezifische Sprachen ermöglichen die Validierung und Optimierung auf Ebene der Problemdomäne.¹¹⁷⁷

Neben den genannten Vorteilen bietet die Verwendung einer DSL einige Nachteile:

- Domänenspezifische Sprachen sind nicht allgemein verfügbar.¹¹⁷⁸
- Die DSL muss spezifiziert, implementiert und getestet werden, was einen Mehraufwand gegenüber der Verwendung allgemeinerer Datenformate bedeutet.¹¹⁷⁹
- Darüber hinaus müssen Abwägungen getroffen werden hinsichtlich der Allgemeinheit der spezifizierten Sprachelemente.¹¹⁸⁰

¹¹⁶⁹ Vgl. VAN DEURSEN et al. (2000), S. 26.

¹¹⁷⁰ Vgl. LERDORF (1995).

¹¹⁷¹ Die Makrosprache in „Excel“ ist ein Beispiel der „Endnutzerprogrammierung“, bei der Endnutzer einer Software einfache Programmierungen vornehmen. Vgl. VAN DEURSEN et al. (2000), S. 27.

¹¹⁷² Vgl. VAN DEURSEN et al. (2000), S. 27.

¹¹⁷³ Vgl. VAN DEURSEN et al. (2000), S. 27.

¹¹⁷⁴ Vgl. VAN DEURSEN et al. (2000), S. 27.

¹¹⁷⁵ Vgl. VAN DEURSEN et al. (2000), S. 27.

¹¹⁷⁶ Vgl. VAN DEURSEN et al. (2000), S. 27.

¹¹⁷⁷ Vgl. VAN DEURSEN et al. (2000), S. 27.

¹¹⁷⁸ Vgl. VAN DEURSEN et al. (2000), S. 27.

¹¹⁷⁹ Vgl. VAN DEURSEN et al. (2000), S. 27.

¹¹⁸⁰ Vgl. VAN DEURSEN et al. (2000), S. 27.

- Zudem sind speziell für eine Problemdomäne entwickelte Sprachen potentiell weniger effizient als problemspezifisch entwickelte Software.¹¹⁸¹

Es lassen sich folgende Feststellungen treffen, die weder als Vor- noch als Nachteil gesehen werden:

- Eine höhere Allgemeingültigkeit wirkt sich positiv auf die Wiederverwendbarkeit der DSL aus, verringert jedoch eventuell ihre Effektivität für die betrachtete Problemdomäne.¹¹⁸²
- Alle Stakeholder der neuen DSL müssen diese nach ihrer Entwicklung erlernen.¹¹⁸³

Listing 3 zeigt beispielhaft eine Nachricht zwischen zwei Agenten in DSL-Notation.

```
message Handshake {
  sender "9A01F36D-9ABD-D4D9-80C7-022AF85C82A8"
  metadata {
    host "127.0.0.1"
    port 15001
    uuid sender
  }
}
```

Listing 3: Beispielhafte Nachricht in DSL-Notation

7.2.3.3 Datenbanksystem

Bedeutung für die Implementierungsarbeit

Ein Datenbanksystem ist nötig, damit die implementierte integrierte Entwicklungsumgebung Daten über Agenten, Haltepunkte, Relationen, Handelspräferenzen und Szenarien bei Bedarf persistent speichern und wieder abfragen kann.¹¹⁸⁴ In der modernen Softwareentwicklung abstrahieren Applikationen den Zugriff auf die Datenbank und schaffen durch Adapter eine Schnittstelle zum Schreiben/Auslesen der Daten, welche die Auswahl des Datenbanksystems zumindest theoretisch relativieren (da die Software durch diese Adapter mit allen gängigen Datenbanksystemen interagieren kann).¹¹⁸⁵

Dennoch lohnt sich die Analyse einiger Alternativen in Bezug auf ihre Erfüllung der genannten Kriterien.

¹¹⁸¹ Vgl. VAN DEURSEN et al. (2000), S. 27.

¹¹⁸² Vgl. VAN DEURSEN et al. (2000), S. 27.

¹¹⁸³ Vgl. VAN DEURSEN et al. (2000), S. 27.

¹¹⁸⁴ REESE et al. beschreiben anhand eines einfachen Beispiels, warum Datenbanksysteme für Webapplikationen wichtig sind. Vgl. REESE et al. (2002), S. 10.

¹¹⁸⁵ Vgl. CARNEIRO/BARAZI (2010), S. 3.

Folgende Datenbanksysteme stehen zur Auswahl.

MySQL

MySQL ist ein relationales Datenbankmanagementsystem (RDBMS).¹¹⁸⁶

Eine Datenbank ist eine geordnete Sammlung von Daten, welche üblicherweise in Dateien abgelegt sind.¹¹⁸⁷ Die Daten sind häufig in Tabellen organisiert und zwischen den Daten unterschiedlicher Tabellen sind Querverweise (Relationen) möglich.¹¹⁸⁸ Ist dies der Fall spricht man von einer relationalen Datenbank.¹¹⁸⁹

RDBMS bedeutet „relational database management system“. ¹¹⁹⁰ Es beschreibt ein System von Applikationen zur Verwaltung relationaler Datenbanken.¹¹⁹¹

Ziele eines RDBMS sind die sichere Speicherung von Daten und das Interpretieren von Kommandos zur Analyse, Manipulation, Sortierung und Selektierung bestehender und neuer Datensätze.¹¹⁹² Beispiele für solche Systeme sind MySQL, Oracle und Microsoft SQL Server.¹¹⁹³ Das Ziel von MySQL ist die strukturierte Speicherung von großen Datenmengen in Tabellen. MySQL ist plattformunabhängig, schnell, sicher und stabil.¹¹⁹⁴

MySQL ist über das Internet kostenlos verfügbar.¹¹⁹⁵

MySQL wird seit über 10 Jahren weiterentwickelt, ausgiebig getestet und dokumentiert.¹¹⁹⁶ MySQL hat sich zu einer der führenden Systeme im Bereich der RDBMS entwickelt. In der Vergangenheit wurde immer wieder die Behauptung veröffentlicht, dass MySQL die meistverwendete Open-Source-Datenbanktechnologie sei.¹¹⁹⁷ Generell sind solche Aussagen jedoch schwierig zu verifizieren oder zu falsifizieren, wie das Beispiel SQLite zeigt.¹¹⁹⁸

Das Konzept von MySQL basiert auf einer Client/Server-Architektur.¹¹⁹⁹ MySQL besitzt einen dedizierten Datenbankserver, der von Clients angesteuert werden kann, bspw.

¹¹⁸⁶ Vgl. KOFLER/KRAMER (2005), S. 4.

¹¹⁸⁷ Vgl. KOFLER/KRAMER (2005), S. 3.

¹¹⁸⁸ Vgl. KOFLER/KRAMER (2005), S. 3.

¹¹⁸⁹ Vgl. KOFLER/KRAMER (2005), S. 3.

¹¹⁹⁰ Vgl. KOFLER/KRAMER (2005), S. 4.

¹¹⁹¹ Vgl. KOFLER/KRAMER (2005), S. 3 f.

¹¹⁹² Vgl. KOFLER/KRAMER (2005), S. 3 f.

¹¹⁹³ Vgl. KOFLER/KRAMER (2005), S. 3 f.

¹¹⁹⁴ Vgl. KOFLER/KRAMER (2005), S. 7 u. 16.

¹¹⁹⁵ Weitere Informationen sind im Internet verfügbar. Vgl. MYSQL (2017).

¹¹⁹⁶ Vgl. KOFLER/KRAMER (2005), S. 7 u. 16.

¹¹⁹⁷ Vgl. MORSY/OTTO (2008), S. 429, sowie KOFLER/KRAMER (2005), S. 16.

¹¹⁹⁸ Vgl. KREIBICH (2010), S. 15 f.

¹¹⁹⁹ Vgl. KOFLER/KRAMER (2005), S. 4.

um Daten zu manipulieren und abzufragen.¹²⁰⁰ Zudem können mehrere MySQL-Server distribuiert betrieben werden, um so die Sicherheit der Daten oder die Ausführungsgeschwindigkeit der Abfragen durch Redundanz zu steigern.¹²⁰¹

Für die Verwendung von MySQL spricht die breite Anwendbarkeit, da es sowohl eine Reihe an Datentypen unterstützt als auch als hochgradig redundantes System aufgesetzt und auf einer breiten Palette an Hardwarekonfigurationen eingesetzt werden kann.¹²⁰²

SQLite

SQLite ist, wie auch MySQL, ein relationales Datenbankmanagementsystem (RDBMS).¹²⁰³ Das Ziel von SQLite ist es, die Konfiguration und den Betrieb eines dedizierten Datenbankservers überflüssig zu machen und das Datenbanksystem stattdessen in die Applikation, die es benutzt, zu integrieren.¹²⁰⁴ Zu diesem Zweck ist SQLite plattformunabhängig, kompakt, verlässlich und muss vor der Inbetriebnahme nicht konfiguriert werden.¹²⁰⁵

SQLite ist über das Internet kostenlos verfügbar.¹²⁰⁶

Auf der Webseite von SQLite wird behauptet, SQLite sei das Datenbanksystem mit den meisten Installationen weltweit. Diese Aussage darf als mutig gelten, wenngleich sie nicht von vorneherein verworfen werden kann.¹²⁰⁷

Eine Stärke von SQLite liegt in der hohen Flexibilität: Der Entwickler muss keinen Datenbankserver konfigurieren und sich keine Sorgen um die Netzwerkkonnektivität machen, da SQLite keine Client-/Server-Architektur besitzt, sondern lokal auf dem Computer ausgeführt wird, auf dem auch die SQLite verwendende Applikation läuft.¹²⁰⁸ Zudem sind keine Lizenzabkommen zu beachten oder Gebühren zu entrichten.¹²⁰⁹ Bei der Ausführungsgeschwindigkeit kann SQLite mit anderen RDBMS mithalten, außer bei sehr komplexen Abfragen („Queries“) auf großen Datenmengen.¹²¹⁰

¹²⁰⁰ Vgl. KOFLER/KRAMER (2005), S. 4 f.

¹²⁰¹ Vgl. KOFLER/KRAMER (2005), S. 4 f., sowie REESE et al. (2002), S. 137 ff.

¹²⁰² Vgl. SCHWARTZ et al. (2008), S. 1.

¹²⁰³ Vgl. KREIBICH (2010), S. 1 ff., sowie OWENS (2006), S. 1.

¹²⁰⁴ Vgl. KREIBICH (2010), S. 1 ff., sowie OWENS (2006), S. 1.

¹²⁰⁵ Vgl. KREIBICH (2010), S. 1 ff., sowie OWENS (2006), S. 1.

¹²⁰⁶ Weitere Informationen sind im Internet verfügbar. Vgl. SQLITE (2017).

¹²⁰⁷ Aufgrund der Tatsache, dass keinerlei Lizenzabkommen zur Nutzung von SQLite nötig sind, kann nicht genau erhoben werden, wieviele installierte SQLite-Versionen es weltweit gibt. Vgl. KREIBICH (2010), S. 15 f.

¹²⁰⁸ Vgl. OWENS (2006), S. 9.

¹²⁰⁹ Vgl. OWENS (2006), S. 9.

¹²¹⁰ Vgl. OWENS (2006), S. 11.

Aufgrund der im Vergleich zu anderen RDBMS einfachen Installation und entfallenden Konfiguration ist SQLite gut geeignet, die Entwicklungszeit einer Webapplikation zu verkürzen. Auch können die erwähnten Nachteile von SQLite im Rahmen eines Entwicklungssystems vernachlässigt werden.¹²¹¹ Es ist allerdings zweifelhaft, ob für einen späteren Produktiveinsatz SQLite zu wählen wäre. Für den produktiven Einsatz wäre ein RDBMS mit dediziertem Datenbankserver, also einem separaten Computer, der nur als Server in einer Client-/Server-Architektur fungiert, aufgrund der höheren Ausführungsgeschwindigkeit besser geeignet.¹²¹²

CouchDB

CouchDB ist ein noch relativ junges, dokumentorientiertes Datenbanksystem. Das Ziel von CouchDB ist, den Anforderungen moderner, webbasierter, dokumentorientierter und distribuerter Applikationen Rechnung zu tragen. Da CouchDB distribuiert betrieben werden kann und in der Programmiersprache Erlang implementiert ist, ist es ein hochgradig verlässlich verfügbares, redundantes und skalierbares Datenbanksystem.¹²¹³

CouchDB ist über das Internet kostenlos verfügbar.¹²¹⁴

Im Gegensatz zu RDBMS ist CouchDB nicht schemabasiert, sondern dokumentorientiert.¹²¹⁵ Dies bedeutet grundsätzlich, dass CouchDB Daten nicht nach einem vordefinierten Tabellenschema speichert, sondern in Form eigenständiger, ganzheitlich voneinander unabhängiger Dokumente (teilweise auch Objekte genannt).¹²¹⁶ CouchDB präsentiert die Informationen im JSON-Format.¹²¹⁷ Eine schemafreie Datenbank hat den Vorteil, dass der Entwickler sich nicht bereits vor der Entwicklung Gedanken zum Aufbau des Tabellenschemas und der damit verbundenen Interdependenzen und Integritätsbedingungen machen muss. Somit begünstigt die Schemafreiheit die Entwicklungszeit einer datenbankbasierten Applikation erheblich. Nachteilig an dieser Konzeption ist aber, dass sie sich in komplexen Applikationen, deren Geschäftslogik auf die Existenz eindeutig definierter Interdependenzen und eines konstanten Integritätsniveaus angewiesen ist, negativ auswirken kann, da die gespeicherten Daten hochgradig unstrukturiert sind.¹²¹⁸ Für diese Fälle ist ein dokumentorientiertes Datenbanksystem weniger geeignet.¹²¹⁹

Ein weiterer Unterschied zu bspw. SQL-Datenbanksystemen ist, dass CouchDB Revisionen der gespeicherten Dokumente archiviert (ähnlich einer Versionsverwaltung, nur dass

¹²¹¹ Vgl. KREIBICH (2010), S. 11 f., sowie KOFLER/KRAMER (2005), S. 7.

¹²¹² Vgl. TATE et al. (2009), S. 153.

¹²¹³ Vgl. LENNON (2009), S. 3 f., sowie WATSON (2009), S. 255 u. 260.

¹²¹⁴ Weitere Informationen sind im Internet verfügbar. Vgl. COUCHDBAPACHE (2017).

¹²¹⁵ Vgl. LENNON (2009), S. 3 f., sowie WATSON (2009), S. 255 u. 260.

¹²¹⁶ Vgl. KOFLER/KRAMER (2005), S. 4.

¹²¹⁷ Vgl. ANDERSON et al. (2010), S. 120, WATSON (2009), S. 255, sowie LENNON (2009), S. 4, 50 u. 87 ff.

¹²¹⁸ Vgl. LENNON (2009), S. 7.

¹²¹⁹ Vgl. LENNON (2009), S. 5 f.

das Datenbanksystem selbst die Versionierung vornimmt).¹²²⁰ Im Gegensatz zu SQL-Datenbanksystemen werden Reports nicht mit Hilfe von SQL-Queries, sondern durch JavaScript-Funktionen erstellt, welche über die in der Datenbank gespeicherten JSON-Dokumente iterieren.¹²²¹ Zur Kommunikation mit dem Datenbankserver stellt CouchDB eine HTTP-API bereit.¹²²² Dies hat den Vorteil, dass jede Plattform, die HTTP-Requests ausführen und JSON interpretieren kann, in der Lage ist, CouchDB als Datenbanksystem einzusetzen.¹²²³

Nicht zuletzt aufgrund der Implementierung in Erlang, der HTTP-API und des Einsatzes von JSON ist CouchDB gut geeignet, im Rahmen einer komplexen Webapplikation, die große Mengen unstrukturierter Daten handhaben muss, als Datenbanksystem eingesetzt zu werden.

7.2.3.4 Webframework

Bedeutung für die Implementierungsarbeit

Ein Webframework ist notwendig, um eine internet-basierte, integrierte Entwicklungsumgebung zu implementieren.

Folgende Webframeworks stehen zur Auswahl.

Sinatra

Sinatra ist ein leichtgewichtiges Webframework, das HTTP-Funktionen für existierende Applikationen bereitstellt und es erlaubt, von Grund auf neue Applikationen so einfach wie möglich zu entwickeln.¹²²⁴

Daher wird Sinatra von seinen Entwicklern auch häufig nicht als Webframework im klassischen Sinn beschrieben, sondern als „DSL zum Erstellen von Webapplikationen“ bezeichnet.¹²²⁵

Sinatra ist über das Internet kostenlos verfügbar.¹²²⁶

Im Gegensatz zu Ruby on Rails forciert Sinatra keine Konzepte, wie bspw. MVC oder REST.¹²²⁷

¹²²⁰ Vgl. LENNON (2009), S. 6.

¹²²¹ Vgl. ANDERSON et al. (2010), S. 13 f., LENNON (2009), S. 7, sowie WATSON (2009), S. 255 f.

¹²²² Vgl. ANDERSON et al. (2010), S. 33 ff.

¹²²³ Vgl. LENNON (2009), S. 7.

¹²²⁴ Vgl. COOPER (2009), S. 387, sowie WATSON (2009), S. 133.

¹²²⁵ Vgl. COOPER (2009), S. 387, sowie WATSON (2009), S. 133.

¹²²⁶ Weitere Informationen sind im Internet verfügbar. Vgl. SINATRA (2017).

¹²²⁷ Vgl. COOPER (2009), S. 388.

Ruby on Rails

Ruby on Rails ist ein in der Programmiersprache Ruby implementiertes Webframework und wurde von David Heinemeier Hansson als Nebenprodukt seiner Arbeit an der Projektmanagementsoftware „Basecamp“ entwickelt.¹²²⁸

Ruby on Rails ist über das Internet kostenlos verfügbar.¹²²⁹

Hansson entschied sich Ruby anstelle von Java zu verwenden und es gelang ihm, das Projekt „Basecamp“ in zwei Monaten zu realisieren, da Ruby ihm das Programmieren so stark vereinfachte, dass er sich auf die Aspekte seines Problems anstatt auf die Aspekte der Programmiersprache konzentrieren musste.¹²³⁰ Er entschloss sich daraufhin, die grundlegende Architektur des Projekts in ein neues Webframework namens „Ruby on Rails“ zu adaptieren, welches die Entwicklung von datenbankgestützten Webapplikationen vereinfachen sollte.¹²³¹

Ruby on Rails folgt dem MVC-Modell, welches zur Beschreibung von Software eingesetzt wird. Es unterteilt eine oberflächenorientierte Software in drei Schichten: Model-, View- und Controller-Schicht. Die Model-Schicht enthält die Datenmodellierung, die View-Schicht die Datenpräsentation und die Controller-Schicht die Programmsteuerung.¹²³²

Ruby on Rails abstrahiert von der Verwendung der Datenbanktechnologie, des JavaScript-Frameworks sowie der Session-Verwaltung, stellt einen Migrationsansatz zur Population der Datenbank zur Verfügung und übernimmt die Verwaltung und Implementierung vieler wiederkehrender Abläufe für den Entwickler.¹²³³

Ruby on Rails hat über die Jahre Entwickler aus den Bereichen Java und C# durch seine einfachere Art der Webentwicklung überzeugt.¹²³⁴ Primär liegt dies am MVC-Modell, den sorgsam gewählten Standardeinstellungen und der Programmiersprache Ruby.¹²³⁵

¹²²⁸ Vgl. CARNEIRO/BARAZI (2010), S. 3, sowie COOPER (2009), S. 104.

¹²²⁹ Weitere Informationen sind im Internet verfügbar. Vgl. RUBYONRAILS (2017).

¹²³⁰ Vgl. COOPER (2009), S. 104.

¹²³¹ Vgl. COOPER (2009), S. 104.

¹²³² Vgl. AMMELBURGER/SCHERER (2008), S. 6 ff.

¹²³³ Die Reduktion der Wiederholung wiederkehrender Abläufe wird primär durch die Paradigmen „convention over configuration“ und „don’t repeat yourself“ gewährleistet. Vgl. CHAK (2009), S. 239, sowie EDIGER (2008), S. 1 u. 2 f.

¹²³⁴ Vgl. CARNEIRO/BARAZI (2010), S. 4.

¹²³⁵ Bei den Standardeinstellungen ist vor allem das Paradigma „convention over configuration“ zu nennen. Vgl. EDIGER (2008), S. 1.

7.2.3.5 Webserver

Bedeutung für die Implementierungsarbeit

Ein Webserver wird benötigt, damit der Softwareprototyp seine Benutzeroberfläche und Schnittstellen über ein Netzwerk, bspw. das Internet, bereitstellen kann.¹²³⁶ Zu diesem Zweck übersetzt ein Webserver eine URL in einen Datei- oder Applikationsnamen und überträgt entweder die betreffende Datei oder die Ausgabe der entsprechenden Applikation.¹²³⁷

Folgende Webserver stehen zur Auswahl.

Apache

Apache ist ein sicherer, schneller Webserver und von den hier beschriebenen Webservern im Funktionsumfang der komplexeste.¹²³⁸

Apache ist über das Internet kostenlos verfügbar.¹²³⁹

Apache ist modular aufgebaut und kann durch entsprechende Erweiterungen komplexe Manipulationen von HTTP-Kopfdaten und URLs durchführen, als Proxyserver eingesetzt werden und die Kommunikation zwischen Browser und Webserver verschlüsseln.¹²⁴⁰

Apache eignet sich prinzipiell zur Entwicklung des Softwareprototyps, wenngleich der Einsatz eines im Funktionsumfang so komplexen Webserver ungerechtfertigt erscheint.¹²⁴¹

WEBrick

WEBrick ist der Standard-Webserver des Ruby-Kernpakets.¹²⁴² WEBrick unterstützt alle Funktionen, die ein Webframework zum Betrieb benötigt, wie bspw. HTTP und HTML, und ist sehr einfach zu konfigurieren.¹²⁴³

WEBrick ist als Teil des Ruby-Kernpakets kostenlos verfügbar.

WEBrick eignet sich wegen des geringeren Konfigurationsaufwands und der damit verbundenen positiven Wirkung auf die Entwicklungszeit zur Entwicklung des Softwareprototyps.

¹²³⁶ Vgl. KERSKEN (2007), S. 273.

¹²³⁷ URL bedeutet „uniform resource locator“ Vgl. KERSKEN (2007), S. 1.

¹²³⁸ Apache ist die gebräuchliche Kurzform für das Produkt „Apache HTTP Server“ der Apache Foundation. Vgl. EDIGER (2008), S. 318, sowie LAURIE/LAURIE (2003), S. 1.

¹²³⁹ Weitere Informationen sind im Internet verfügbar. Vgl. APACHE (2017).

¹²⁴⁰ Vgl. LAURIE/LAURIE (2003), S. 84, 179, 188 u. 230.

¹²⁴¹ Wenn sie die Wahl haben, entscheiden sich Administratoren häufig für leichtgewichtiger Lösungen. Vgl. EDIGER (2008), S. 318.

¹²⁴² Vgl. CARLSON/RICHARDSON (2006), S. 546.

¹²⁴³ Vgl. CARLSON/RICHARDSON (2006), S. 547.

7.2.3.6 Versionsverwaltung

Bedeutung für die Implementierungsarbeit

Im Rahmen der Versionsverwaltung geht es darum, Änderungen an einem Quelltext zu erfassen und eine mit Zeitstempeln archivierte Historie aller hierfür vorgesehenen Dateien zu führen. Der Vorgang des Hinzufügens von Änderungen wird in der Literatur als „einchecken“ bezeichnet. Das Archiv, das alle eingeecheckten Änderungen enthält, wird im Folgenden als „Repository“ bezeichnet.¹²⁴⁴ So können Änderungen, die vom Entwickler im Nachhinein als falsch, schadhaft oder anderweitig unvorteilhaft identifiziert werden, zurückgenommen werden, indem aus dem Archiv der Versionsverwaltung eine ältere, als besser erachtete Version der betreffenden Dateien wiederhergestellt wird.¹²⁴⁵

Folgende Versionsverwaltungen stehen zur Auswahl.

Git

Git ist eine distribuierte Versionsverwaltung zur Verwaltung der Änderungen an Verzeichnissen.¹²⁴⁶ Ziel von Git ist es, die distribuierte, unabhängige und gleichzeitige Entwicklung in privaten Repositories zu ermöglichen und zu fördern.¹²⁴⁷ Git lehnt sich an Konzepte von Bitkeeper und Mercurial an.¹²⁴⁸

Git ist über das Internet kostenlos verfügbar.¹²⁴⁹

In Git sind Transaktionen atomar und einmal eingeecheckte Daten unveränderbar.¹²⁵⁰ Git ist sowohl beim „branching“ und „merging“ als auch beim Übertragen der eingeecheckten Änderungen über das Netzwerk sehr schnell und effizient.¹²⁵¹ Hierdurch können bspw. Ideen für neue Funktionalitäten schnell und unabhängig voneinander (und unabhängig vom Hauptentwicklungszweig) ausprobiert, evaluiert und in andere Entwicklungszweige übernommen oder verworfen werden.¹²⁵²

Git ist gut geeignet, die Entwicklung eines Softwareprototyps zu unterstützen, da Git gegenüber anderen gängigen und kostenlosen Versionsverwaltungen entscheidende Vorteile bei der schnellen, flexiblen Entwicklung von Software bietet. Die entscheidenden Vorteile

¹²⁴⁴ Vgl. PILATO et al. (2008), S. 1.

¹²⁴⁵ Vgl. CHACON (2014), S. 1, sowie BERLIN/ROONEY (2006), S. 1.

¹²⁴⁶ Vgl. LOELIGER/MCCULLOUGH (2012), S. 1.

¹²⁴⁷ Vgl. LOELIGER/MCCULLOUGH (2012), S. 2.

¹²⁴⁸ Vgl. LOELIGER/MCCULLOUGH (2012), S. 2 f.

¹²⁴⁹ Weitere Informationen sind im Internet verfügbar. Vgl. GITSCM (2017).

¹²⁵⁰ Vgl. LOELIGER/MCCULLOUGH (2012), S. 3.

¹²⁵¹ Vgl. LOELIGER/MCCULLOUGH (2012), S. 2 f.

¹²⁵² Vgl. LOELIGER/MCCULLOUGH (2012), S. 89 f.

sind die Erleichterung des distribuierten Arbeitens durch einfaches und schnelles „branching“ und „merging“ sowie die Tatsache, dass jede Arbeitskopie eine komplette Historie aller Entwicklungszweige enthält.¹²⁵³

SVN

Subversion (SVN) ist eine zentralisierte Versionsverwaltung zur Verwaltung der Änderungen an einzelnen Dateien.¹²⁵⁴ Das Konzept von SVN lehnt sich stark an das von Concurrent Versions System (CVS) an. CVS ist ebenfalls eine Versionsverwaltung.¹²⁵⁵ Das Ziel von SVN ist es, die Fehler, die bei der Konzeption von CVS gemacht wurden, nicht zu wiederholen.¹²⁵⁶ Aus diesem Grund ähneln sich Arbeitsabläufe unter CVS und SVN stark, was nicht zuletzt dazu beigetragen hat, dass viele unzufriedene CVS-Benutzer zu SVN wechselten.¹²⁵⁷

SVN ist über das Internet kostenlos verfügbar.¹²⁵⁸

SVN speichert Änderungen atomar, bietet Unterstützung von binären Dateien und erlaubt das Abspalten einzelner Entwicklungszweige und -versionen mittels „branching“ und „tagging“.¹²⁵⁹ SVN speichert die eingetragenen Änderungen zentral auf einem Server, auf den alle Entwickler eines Projekts Zugriff haben können.¹²⁶⁰ SVN ist kein SCM, also keine Versionsverwaltung, die speziell für das Archivieren von Quelltexten entwickelt wurde. SCM bedeutet „software configuration management“.¹²⁶¹ SCM bietet einige Vorteile gegenüber reiner Versionsverwaltung, da sie den eingetragenen Quelltext auch syntaktisch versteht und Werkzeuge, bspw. zu dessen Kompilierung und Distribution, bereitstellt.¹²⁶²

SVN ist durchaus geeignet, die Entwicklung eines Softwareprototyps zu unterstützen. Allerdings wirkt sich der, insbesondere im Vergleich zu Git, höhere Konfigurationsaufwand negativ auf die Entwicklungszeit aus. Der Grund hierfür ist, dass für SVN ein dedizierter Server konfiguriert und betrieben werden muss.¹²⁶³

¹²⁵³ Vgl. CHACON (2014), S. 5 ff., sowie LOELIGER/MCCULLOUGH (2012), S. 3 f.

¹²⁵⁴ Vgl. PILATO et al. (2008), S. 318.

¹²⁵⁵ Vgl. PILATO et al. (2008), S. 19.

¹²⁵⁶ Vgl. PILATO et al. (2008), S. 19.

¹²⁵⁷ Vgl. BERLIN/ROONEY (2006), S. 5.

¹²⁵⁸ Weitere Informationen sind im Internet verfügbar. Vgl. SUBVERSION (2017).

¹²⁵⁹ Vgl. BERLIN/ROONEY (2006), S. 4 f.

¹²⁶⁰ Vgl. BERLIN/ROONEY (2006), S. 4.

¹²⁶¹ Vgl. PILATO et al. (2008), S. xiii.

¹²⁶² Vgl. PILATO et al. (2008), S. xiii.

¹²⁶³ Vgl. PILATO et al. (2008), S. 191 ff., sowie BERLIN/ROONEY (2006), S. 4.

7.2.3.7 Browsertechnologien

Beschreibung

Wie in Kapitel 7.2.1 gezeigt wurde, gibt es zu den vorgestellten Browsertechnologien keine wirtschaftlich einsetzbaren Alternativen.¹²⁶⁴ Da folglich bei den Browsertechnologien keine direkte Auswahlproblematik besteht, werden im Folgenden die eingesetzten Browsertechnologien beschrieben.

HTML

HTML ist eine Auszeichnungssprache, mit deren Hilfe Hypertext-Dokumente durch Auszeichnungen angereichert werden, so dass der reine Text des Dokuments eine weitere Ebene an Metadaten erhält.¹²⁶⁵

Beispiele für diese Art von Anreicherung sind:

- **Text-Formatierungen:** Textteile können mit Hilfe von Auszeichnungen formatiert werden, so dass sie bspw. als kursiv, fett oder unterstrichen dargestellt werden.
- **Links:** Durch Links wird der Browser angewiesen, einen bestimmten Textteil klickbar zu machen, so dass der Benutzer bei einem Klick auf diesen Textteil auf eine andere Webseite geleitet wird.
- **Bildverweise:** HTML-Dokumente enthalten lediglich Verweise auf Bilder, nicht jedoch die Bilder selbst. Dadurch, dass diese Bildverweise ausgezeichnet sind, wird der Browser angewiesen, das Bild herunterzuladen und darzustellen.

HTML dient der Strukturierung von Informationen in Dokumenten und wird vom World Wide Web Consortium (W3C) weiterentwickelt. Das W3C ist eine der Standardisierungsorganisationen des Internets.¹²⁶⁶ HTML-Dokumente enthalten sowohl sichtbare Informationen wie die angeführten Beispiele als auch unsichtbare Metainformationen, welche für den Browser oder Suchmaschinen wie bspw. Google relevant sind.¹²⁶⁷

HTML-Dateien enthalten idealerweise nur die Struktur eines Dokuments und stellen qualitativ hochwertige Informationen zur Verfügung, während die Präsentation dieser Informationen der CSS-Technologie überlassen wird.¹²⁶⁸

Die Anwendung von HTML geht heute über die Darstellung klassischer Webseiten hinaus. So werden immer häufiger Programmkomponenten durch eine Kombination von HTML

¹²⁶⁴ Vgl. Kapitel 7.2.1, S. 156.

¹²⁶⁵ Vgl. MUSCIANO/KENNEDY (2007), S. 8.

¹²⁶⁶ Vgl. MUSCIANO/KENNEDY (2007), S. 7 f.

¹²⁶⁷ Vgl. MUSCIANO/KENNEDY (2007), S. 8 u. 208 ff.

¹²⁶⁸ Vgl. MUSCIANO/KENNEDY (2007), S. 9 u. 230.

und CSS dargestellt.¹²⁶⁹ Auch im Bereich der Mobilgeräte gewinnt HTML zunehmend an Bedeutung.¹²⁷⁰

CSS

CSS ist eine Technologie, um Informationen, bspw. in HTML-Seiten, zu visualisieren.¹²⁷¹ Dabei geht es primär um die ansprechende visuelle Aufbereitung der Inhalte durch Farben, Formen und Grafiken, aber auch um die Barrierefreiheit angebotener Inhalte, wie bspw. die Lesbarkeit von Webseiten durch sogenannte Screenreader.¹²⁷²

Bevor das W3C 1996 das erste Konzept zu CSS veröffentlichte, mussten alle visuellen Eigenschaften von HTML-Elementen in den Elementen selbst definiert werden.¹²⁷³ Mit der Einführung von CSS wurde es möglich, allgemeine Regeln zur Darstellung der HTML-Elemente zu definieren. Bspw. konnte nun festgelegt werden, dass alle Überschriften grau dargestellt werden sollen.¹²⁷⁴

Ein entscheidender Grundgedanke von CSS ist es, die Präsentationsinformationen zu den in HTML-Dateien vorhandenen Daten von den HTML-Dateien selbst zu trennen und an eine andere Stelle auszulagern.¹²⁷⁵ Das Konzept von CSS erreicht reichhaltigere Gestaltungsmöglichkeiten und geringere Dateigrößen durch seine kaskadierende Natur.¹²⁷⁶

Die Anwendung von CSS geht heute weit über den Einsatz zur Visualisierung von Webseiten hinaus. Teilweise stellen auch Applikationen, wie bspw. der Mozilla-Browser, ihre eigenen Programmkomponenten mit Hilfe von CSS dar.¹²⁷⁷

JavaScript

JavaScript ist eine Skriptsprache, welche dem Benutzer die Interaktion mit sonst statischen Webseiten ermöglicht und als „Sprache des Internets“ bezeichnet wird.¹²⁷⁸ JavaScript vereinfacht die Interaktion mit Webseiten, indem es Manipulationen des sonst statischen Inhalts ermöglicht.¹²⁷⁹ So wird es möglich, auf Benutzerinteraktionen und andere Ereignis-

¹²⁶⁹ Vgl. MEYER (2007), S. 1.

¹²⁷⁰ Vgl. MUSCIANO/KENNEDY (2007), S. 457 ff.

¹²⁷¹ Vgl. ANDREW/SHAFFER (2006), S. 3 ff. u. 23 ff.

¹²⁷² Screenreader sind Programme, die Webseiten vorlesen (bspw. für Menschen mit Sehschwäche).

¹²⁷³ Bspw. musste in jeder einzelnen Überschrift explizit definiert werden, welche Farbe die jeweilige Überschrift haben sollte. Vgl. MUSCIANO/KENNEDY (2007), S. 230.

¹²⁷⁴ Vgl. MUSCIANO/KENNEDY (2007), S. 230.

¹²⁷⁵ Vgl. MEYER (2007), S. 3 ff.

¹²⁷⁶ Vgl. MEYER (2007), S. 3 ff.

¹²⁷⁷ Vgl. MEYER (2007), S. 1.

¹²⁷⁸ Vgl. CROCKFORD (2008), S. 1.

¹²⁷⁹ Vgl. FLANAGAN (2011), S. 1 u. 4 ff.

nisse zu reagieren (bspw. durch Öffnen eines Popup-Fensters),¹²⁸⁰ Informationen auf dem Rechner des Benutzers zu speichern¹²⁸¹ und mit Formularen zu interagieren.¹²⁸² Es existieren keine Alternativen zum Einsatz von JavaScript, da JavaScript als einzige Sprache in allen modernen Webbrowsern implementiert ist.¹²⁸³

Das Einsatzgebiet von JavaScript geht heute, ähnlich dem von HTML und CSS, weit über die Implementierung klassischer Webseiten hinaus. So lassen sich mittlerweile aufgrund stetig leistungsfähiger werdender Interpreter vollwertige Applikationen, wie bspw. Webserver, mit JavaScript realisieren.¹²⁸⁴

7.2.4 Auswahl von Implementierungstechniken

7.2.4.1 Vorbemerkungen

Die Bewertung der ausgewählten Implementierungstechniken wird subjektiv durch den Autor auf Basis seiner langjährigen, professionellen Erfahrungen vorgenommen und dient der Einordnung der Implementierungstechniken in Bezug auf die in Kapitel 7.2.2 identifizierten Bewertungskriterien.¹²⁸⁵ Während des Auswahlprozesses dominierte keine der berücksichtigten Implementierungstechniken alle anderen Alternativen in der jeweiligen Domäne (Agentenprogrammiersprache, Datenformat etc.).

Aus Gründen der Vollständigkeit werden sowohl primäre als auch sekundäre Kriterien aufgeführt. Von den sechs identifizierten Bewertungskriterien werden im Auswahlprozess lediglich fünf betrachtet. Das Kriterium der „geringen Kosten“ wird nicht aufgeführt, da seine Ausprägung konstant und damit für die Bewertung irrelevant ist.

Die vorgestellten Alternativen sind allesamt als geeignete Implementierungstechniken für die zu erfüllende Aufgabe anzusehen. Daher stellt die jeweils gewählte Alternative lediglich eine nicht dominierte Alternative dar, welche in der jeweiligen Domäne aufgrund der Aggregation der Ausprägungen der drei primären Bewertungskriterien die relativ vorteilhafteste Auswahl darstellt. Die beiden sekundären Kriterien werden aus Gründen der Vollständigkeit in den tabellarischen Darstellungen separat aufgeführt. Da die Problem-domäne dieser Forschungsarbeit sehr weit ist und die mit Hilfe der Implementierungstechniken durchgeführten Arbeiten sehr innovativ sind, erheben die in Bezug auf die Implementierungstechniken erstellten Bewertungen keinen streng wissenschaftlichen Anspruch auf Objektivität, Reliabilität und Validität.

¹²⁸⁰ Vgl. FLANAGAN (2011), S. 445 ff.

¹²⁸¹ Vgl. FLANAGAN (2011), S. 449 ff.

¹²⁸² Vgl. FLANAGAN (2011), S. 400 ff.

¹²⁸³ Vgl. CROCKFORD (2008), S. 4.

¹²⁸⁴ Vgl. STEFANOV (2010), S. 1.

¹²⁸⁵ Vgl. Kapitel 7.2.2, S. 157.

7.2.4.2 Agentenprogrammiersprache

Tabelle 7 stellt eine Übersicht und Beurteilung der vorgestellten Agentenprogrammiersprachen dar. Die Beurteilung dieser und aller weiterer Implementierungstechniken erfolgt auf Basis der in Kapitel 7.2.1 dargelegten Eigenschaften der jeweiligen Agentenprogrammiersprache sowie der sich ergebenden Vor- und Nachteile für die schnelle Implementierung eines Softwareprototyps.

Programmiersprache						
	geringe Kosten	geringe Entwicklungszeit	hohe Flexibilität	hohe Ausführungsgeschwindigkeit	hohe Sicherheit	
						++ sehr gute Eignung
						+ gute Eignung
						- geringe Eignung
						-- schlechte Eignung
Ruby	++	++	++	-	-	
Python	++	++	++	-	-	
C#	++	--	-	++	+	
Java	++	--	-	++	+	

Tabelle 7: Beurteilung von Agentenprogrammiersprachen bezüglich ihrer Eignung zur Implementierung des Softwareprototyps

Ruby und Python sind quelloffen und werden für Windows und unixoide Systeme bereitgestellt. C# wird dagegen offiziell nur für Windows zur Verfügung gestellt, so dass die Plattformunabhängigkeit eingeschränkt ist.¹²⁸⁶

Ruby und Python sind die flexibelsten der beschriebenen Agentenprogrammiersprachen, wodurch eine schnelle, elegante und vor allem problemorientierte Programmierung unterstützt wird und Konzepte der Metaprogrammierung umgesetzt werden können.

Ruby und Python sind in der Ausführung bestimmter Operationen nicht so performant wie Java und C# und bieten ein geringeres Niveau an Sicherheit. Beide Probleme werden bei der Entwicklung eines Softwareprototyps jedoch als weniger relevant erachtet als sie es bspw. bei der Konzeption und Implementierung einer Software für den Produktiveinsatz

¹²⁸⁶ Es existieren Open-Source-Projekte, die C# auch für unixoide Systeme zur Verfügung stellen. Vgl. Kapitel 7.2.3.1, S. 161.

wären.¹²⁸⁷ Ruby und Python bieten zudem Möglichkeiten, besonders lastsensible Teile einer Applikation durch die Programmierung sogenannter „C-Extensions“ zu beschleunigen.¹²⁸⁸

Ruby und Python stellen als Agentenprogrammiersprachen somit nicht dominierte Alternativen zur schnellen prototypischen Implementierung eines Multi-Agenten-Systems und komplementärer internetbasierter Software dar. Da Ruby im Bereich der Entwicklung internetbasierter Systeme bessere Anbindungen besitzt als Python, wird Ruby als primäre Agentenprogrammiersprache gewählt. Aufgrund der besseren Anbindung an sog. Solver-Technologie¹²⁸⁹ wird zur Implementierung des Auktionsmoduls Python gewählt.

7.2.4.3 Datenformat

Tabelle 8 stellt eine Übersicht und Beurteilung der vorgestellten Datenformate dar.

Datenformat	geringe Kosten geringe Entwicklungszeit hohe Flexibilität			hohe Ausführungsgeschwindigkeit hohe Sicherheit		++ sehr gute Eignung + gute Eignung - geringe Eignung - - schlechte Eignung
	geringe Kosten	geringe Entwicklungszeit	hohe Flexibilität	hohe Ausführungsgeschwindigkeit	hohe Sicherheit	
XML	++	-	-	+	++	
JSON	++	+	+	+	-	
DSL	++	+	++	+	++	

Tabelle 8: Beurteilung von Datenformaten bezüglich ihrer Eignung zur Implementierung des Softwareprototyps

Eine DSL verspricht einen dem jeweiligen Problem angemessenen Abstraktionsgrad bei der Interoperabilität mit beliebigen Datenformaten, wenngleich die Entwicklung domänenspezifischer Sprachen einen Mehraufwand bei der Konzeption und Implementierung des Softwareprototyps bedeutet.¹²⁹⁰ Dieser Nachteil erscheint jedoch akzeptabel, da sich

¹²⁸⁷ Vgl. CARLSON/RICHARDSON (2006), S. 817 ff.

¹²⁸⁸ Vgl. CARLSON/RICHARDSON (2006), S. 817 ff., sowie MARTELLI (2006), S. 614 ff.

¹²⁸⁹ Vgl. Kapitel 7.4.3.5, S. 227.

¹²⁹⁰ Vgl. VAN DEURSEN/KLINT (2002), S. 1.

die gewählte Programmiersprache Ruby gut zur Implementierung domänenspezifischer Sprachen eignet.¹²⁹¹

Die Schaffung einer DSL begünstigt bei Betrachtung der gesamten Projektdauer eine geringe Entwicklungszeit, wenngleich JSON schnellere erste Ergebnisse verspricht, da für alle populären Programmiersprachen JSON-Anbindungen existieren.¹²⁹² Die Auswahl von domänenspezifischen Sprachen (DSL) bietet jedoch den Vorteil, dass alle Daten zu einem späteren Zeitpunkt in ein geeignetes Datenformat wie JSON oder XML konvertiert werden können, sollte dies notwendig werden. Dieser Vorteil ist nicht zu unterschätzen, da die spätere Konvertierung zwischen Datenformaten ein nichttriviales Problem darstellt.¹²⁹³

Für die im Rahmen dieser Forschungsarbeit zu bearbeitenden Datenmengen ist eine gleich hohe Ausführungsgeschwindigkeit für alle Alternativen zu erwarten.

Da innerhalb der DSL die problemspezifischen Daten mit den Begriffen der Problemdomäne beschrieben werden können, bietet sie zudem die größte Flexibilität. Aus dem selben Grund bietet die DSL eine hohe Sicherheit, da eine Eingabe invalider Daten durch die Regeln der domänenspezifischen Sprache verhindert wird. Die gleiche Sicherheit kann in XML theoretisch durch Verwendung einer Schemasprache erreicht werden.¹²⁹⁴

Eine DSL stellt als Datenformat somit eine nicht dominierte Alternative zur schnellen prototypischen Implementierung eines Multi-Agenten-Systems dar.

7.2.4.4 Datenbanksystem

Tabelle 9 stellt eine Übersicht und Beurteilung der vorgestellten Datenbanksysteme dar.

Die beschriebenen RDBMS MySQL und SQLite bieten für die Entwicklung des Softwareprototyps die gleichen Basisfunktionen. MySQL ist in bestimmten Fällen schneller, allerdings auch aufwendiger zu konfigurieren als SQLite und erfordert zusätzlich den Betrieb eines dedizierten Servers.

Das dokumentorientierte Datenbanksystem CouchDB ist für die Entwicklung des Softwareprototyps weniger gut geeignet als ein RDBMS, da ein tabellenorientiertes Konzept durch die im Vorfeld notwendige Definition der Daten und ihrer Typen sowie Beziehungen für den Prototyp einer Online-Frachtenbörse eine höhere Sicherheit bietet als ein dokumentorientierte Konzept.

¹²⁹¹ Vgl. GAMBLE et al. (2010), S. 4.

¹²⁹² Vgl. Kapitel 7.2.3.2, S. 167.

¹²⁹³ So existieren bspw. zahlreiche Verfahren zur Konvertierung von XML nach JSON, ohne dass sich ein Standardverfahren durchsetzen konnte. Beispiele für solche Verfahren sind JSONx, JsonML, XSLTJSON, sowie MLJSON. Vgl. LEE (2011), S. 5.

¹²⁹⁴ Vgl. Kapitel 7.2.3.2, S. 166.

Datenbanksystem	geringe Kosten	geringe Entwicklungszeit	hohe Flexibilität	hohe Ausführungsgeschwindigkeit	hohe Sicherheit	
						++ sehr gute Eignung
						+ gute Eignung
						- geringe Eignung
						-- schlechte Eignung
SQLite	++	++	+	+	+	
MySQL	++	+	+	++	+	
CouchDB	++	-	+	-	+	

Tabelle 9: Beurteilung von Datenbanksystemen bezüglich ihrer Eignung zur Implementierung des Softwareprototyps

Für einen Produktiveinsatz würde MySQL aufgrund seiner flexiblen Einsetzbarkeit und weiten Verbreitung bevorzugt.¹²⁹⁵ Jedoch ist die Konfiguration des MySQL-Servers im Vergleich zu SQLite aufwendig, während SQLite für die Implementierung eines Softwareprototyps die gleichen Basisfunktionen bereitstellt wie MySQL und damit eine kurze Entwicklungszeit begünstigt.

SQLite stellt als Datenbanksystem somit eine nicht dominierte Alternative zur schnellen prototypischen Implementierung komplementärer internetbasierter Software für das Multi-Agenten-System dar.

7.2.4.5 Webframework

Tabelle 10 stellt eine Übersicht und Beurteilung der vorgestellten Webframeworks dar.

Ruby on Rails führt Grundideen von Ruby konsequent fort und bietet eine Vielzahl an vorgefertigten Lösungen für standardisierte Probleme, was eine kurze Entwicklungszeit begünstigt.¹²⁹⁶

Ruby on Rails ist weniger flexibel als Sinatra, allerdings wird dieser Nachteil durch die sonst bessere Eignung zur schnellen Implementierung des Softwareprototyps vernachlässigbar.¹²⁹⁷ Ruby on Rails gibt einige Standardverfahren und Paradigmen vor, die

¹²⁹⁵ Vgl. Kapitel 7.2.3.3, S. 172.

¹²⁹⁶ Vgl. CARNEIRO/BARAZI (2010), S. 4 ff.

¹²⁹⁷ Vgl. CARNEIRO/BARAZI (2010), S. 4 ff.

Webframework	geringe Kosten	geringe Entwicklungszeit	hohe Flexibilität	hohe Ausführungsgeschwindigkeit	hohe Sicherheit	
						++ sehr gute Eignung
						+ gute Eignung
						- geringe Eignung
						-- schlechte Eignung
Ruby on Rails	++	++	+	-	+	
Sinatra	++	--	+	+	+	

Tabelle 10: Beurteilung von Webframeworks bezüglich ihrer Eignung zur Implementierung des Softwareprototyps

als inflexibel charakterisiert werden können.¹²⁹⁸ Allerdings begünstigen diese Paradigmen die schnelle Implementierung einer komplexen, datenbankgestützten Webapplikation.¹²⁹⁹

Sinatra ist ein im Kern schlankes Webframework, dessen produktiver Einsatz jedoch bei der Implementierung einer grafischen Benutzeroberfläche Vorarbeit erfordert, um alle benötigten Zusatzkomponenten zu identifizieren und zu installieren. Es eignet sich eher zur Integration von Schnittstellen zur Anbindung von Drittanbietern als zur schnellen Umsetzung komplexer Prototypen mit einer grafischen Benutzeroberfläche. Installiert der Entwickler alle benötigten Zusatzkomponenten zur Erreichung der funktionalen Parität mit Ruby on Rails, geht der Vorteil bei der Ausführungsgeschwindigkeit verloren.

Ruby on Rails und Sinatra sind in gleichem Maße sicher.

Ruby on Rails stellt als Webframework somit eine nicht dominierte Alternative zur schnellen prototypischen Implementierung komplementärer internetbasierter Software für das Multi-Agenten-System dar.

¹²⁹⁸ Vgl. CARNEIRO/BARAZI (2010), S. 4 ff.

¹²⁹⁹ Vgl. CARNEIRO/BARAZI (2010), S. 4 ff.

7.2.4.6 Webserver

Tabelle 11 stellt eine Übersicht und Beurteilung der vorgestellten Webserver dar.

Webserver	geringe Kosten	geringe Entwicklungszeit	hohe Flexibilität	hohe Ausführungsgeschwindigkeit	hohe Sicherheit	
						++ sehr gute Eignung
						+ gute Eignung
						- geringe Eignung
						-- schlechte Eignung
WEBrick	++	++	+	-	-	
Apache	++	-	+	++	+	

Tabelle 11: Beurteilung von Webservern bezüglich ihrer Eignung zur Implementierung des Softwareprototyps

Alle Webserver bieten für die Entwicklung des Softwareprototyps die gleichen Basisfunktionen. Apache ist schneller als WEBrick, erfordert allerdings eine umfangreiche Konfiguration des Servers.

WEBrick ist im Vergleich weniger komplex und sehr einfach zu konfigurieren, was die möglichst schnelle Implementierung eines Softwareprototyps unterstützt.

WEBrick stellt als Webserver somit eine nicht dominierte Alternative zur schnellen prototypischen Implementierung komplementärer internetbasierter Software für das Multi-Agenten-System dar.

7.2.4.7 Versionsverwaltung

Tabelle 12 stellt eine Übersicht und Beurteilung der vorgestellten Software zur Versionsverwaltung dar.

Die hohe Flexibilität von Git im Bereich „branching and merging“ und die Schnelligkeit bei der Ausführung dieser Operationen begünstigen eine schnelle Entwicklungszeit, da so während der Entwicklung leicht Ideen ausprobiert und anschließend mühelos verworfen oder integriert werden können. SVN bietet diese Funktionalitäten ebenfalls, allerdings ist

Versionsverwaltung	geringe Kosten	geringe Entwicklungszeit	hohe Flexibilität	hohe Ausführungsgeschwindigkeit	hohe Sicherheit	
						++ sehr gute Eignung
						+ gute Eignung
						- geringe Eignung
						-- schlechte Eignung
Git	++	++	+	++	++	
SVN	++	+	+	-	-	

Tabelle 12: Beurteilung von Software zur Versionsverwaltung bezüglich ihrer Eignung zur Implementierung des Softwareprototyps

die Nutzung aufwendiger und teilweise weniger performant. Es gibt in SVN kein natives „branching“. „Merging“ ist ein umständlicher, langsamer Prozess, der häufig manuell durchgeführt werden muss.¹³⁰⁰

Ebenfalls für Git spricht die Tatsache, dass kein Server betrieben werden muss, um ein Repository anzulegen und Änderungen einzuchecken, aber ein Server betrieben werden kann, wenn mehrere Entwickler kollaborativ an dem selben Quelltext arbeiten möchten. So kann Git auch ohne Netzwerkkonnektivität von einem einzelnen Computer aus vollumfänglich genutzt werden, da jede Arbeitskopie die komplette Historie aller Entwicklungszweige eines Repositories enthält. Diese distribuierte Natur der Repositories sorgt für Redundanz und damit für eine hohe Sicherheit, da bspw. drei Entwickler, die Git aus Gründen des Datenabgleichs über einen zentralen Server gemeinsam nutzen, vier Kopien der gesamten Entwicklungshistorie des Projekts besitzen. Im Falle von SVN gäbe es zunächst nur den zentralen Server. Um die gleiche Redundanz zu erreichen, müsste ein Administrator drei weitere Server als Backups konfigurieren. Auch dieser Umstand zeigt die höhere Flexibilität von Git.

Git dominiert SVN als Software zur Versionsverwaltung und wird als Implementierungstechnik gewählt.

¹³⁰⁰ Vgl. PILATO et al. (2008), S. 48 ff. u. 59 ff.

7.3 Implementierung der integrierten Entwicklungsumgebung

7.3.1 Motivation

Um verschiedene Agentenkonfigurationen und Implementierungen des entwickelten Verkehrsnetzes verwalten zu können, wird zunächst eine integrierte Entwicklungsumgebung (IDE¹³⁰¹) implementiert, die es ermöglichen soll, die prototypische Implementierung autonomer Software-Agenten in verschiedenen Szenarien testen zu können.

Die Idee hinter der Einführung einer IDE ist die Bereitstellung von Informationen mehrerer Werkzeuge, insbesondere wenn diese der Entwicklung derselben Sache dienen, um gedoppelte und inkonsistent implementierte Funktionalitäten zu vermeiden.

Aus technischer Sicht ist eine IDE zur Implementierung und zum späteren Betrieb der Agenten-Software folglich nicht notwendig. Das beschriebene Multi-Agenten-System könnte auch implementiert werden, indem die später in Kapitel 7.4 beschriebenen Softwaremodule zusammen mit einer individuellen Benutzeroberfläche implementiert werden. Anschließend müssten mehrere Instanzen dieser Software installiert und manuell zu Versuchszwecken gestartet werden. Die einzelnen Instanzen der Software halten ihre Aktionen und insbesondere den Ausgang der Auktionen in sog. Log-Dateien fest, was eine spätere manuelle Auswertung der Ergebnisse möglich machen würde.

Die Entwicklung einer IDE wird dadurch begründet, dass eine integrierte Entwicklungsumgebung mit grafischer Benutzeroberfläche die schnelle Entwicklung verschiedener Szenarien unterstützt, eine bessere Übersicht bei der Szenariobildung („Versuchsanordnung“) bietet, eine unmittelbare Auswertung der während einer Simulation anfallenden Daten ermöglicht und die Vergleichbarkeit verschiedener Szenarien erhöht.

Sie verknüpft Werkzeuge zur Projektverwaltung, zur grafischen Erstellung von Verkehrsnetzen, zur Verwaltung von Agentenkonfigurationen und zur Auswertung der von den Agenten sonst in Log-Dateien festgehaltenen Aktionen. So können innerhalb der Entwicklungsumgebung Projekte mit projektspezifischen Verkehrsnetzen und Agentenkonfigurationen angelegt werden. Innerhalb eines Projekts können Agentenkonfigurationen mit Teilen des Verkehrsnetzes zu Szenarien verknüpft werden. Die Entwicklungsumgebung ermöglicht die Simulation dieser Szenarien. Die durchlaufenen Simulationen können mit Hilfe der Benutzeroberfläche der Entwicklungsumgebung gestartet, beendet, ausgewertet und wiederholt werden.

Durch diese Unterstützung der Vorbereitung, Durchführung und Analyse der im Rahmen dieser Forschungsarbeit notwendigen Experimente mit der implementierten Agenten-Software trägt eine IDE in entscheidendem Maße nicht nur zur Implementierung der Agenten-Software, sondern auch zur Nachprüfbarkeit der erzielten Ergebnisse und gewonnenen Erkenntnisse bei.

¹³⁰¹ Im Englischen „integrated development environment“. Vgl. ERDWEIG et al. (2014), S. 70.

7.3.2 Ziele der Implementierung der Entwicklungsumgebung

Die Unterstützung des Entwicklungsprozesses der prototypischen Implementierung von autonomen Software-Agenten ist das primäre Ziel der Implementierung der Entwicklungsumgebung.

Neben diesem Primärziel verfolgt die Implementierung weitere Ziele, welche das Primärziel unterstützen. Diese sekundären Ziele lassen sich unterteilen in Ziele, welche die Erstellung von Testumgebungen unterstützen, und solche, welche die Auswertung von Simulationen in diesen Testumgebungen unterstützen.

- Die Möglichkeit zur Reduzierung, Erweiterung oder Neugestaltung des zugrundeliegenden Verkehrsnetzes mit Hilfe einer grafischen Benutzeroberfläche,
- die Möglichkeit zur Erstellung beliebig vieler Agentenkonfigurationen mit individuellen Präferenzen,
- die Erleichterung des Anlegens neuer Testumgebungen in Form von Szenarien mit individuellen Verkehrsnetzen und Agentenkonfigurationen,
- die Möglichkeit zur Durchführung von Simulationen ohne Programmierkenntnisse innerhalb der Entwicklungsumgebung sowie
- die Möglichkeit zur analytischen und grafischen Auswertung der durchgeführten Simulationen.

7.3.3 Leistungsumfang

7.3.3.1 Benutzeroberfläche

Die Benutzeroberfläche der Entwicklungsumgebung ist webbasiert. Der Grund hierfür ist die hohe Zugänglichkeit von webbasierten Benutzeroberflächen, die durch eine breite Auswahl von Endgeräten unabhängig von deren Betriebssystemen bedient werden können. Der Zugriff ist von einem PC aus möglich unabhängig davon, ob als Betriebssystem Microsoft Windows, Apple Mac OS X oder Linux installiert ist. Auch der Zugriff von Smartphones oder Tablets wird hierdurch möglich. Diese Vorgehensweise bietet zudem den Vorteil, dass – gegenüber der alternativen Vorgehensweise, einen Prototyp bspw. in Java zu implementieren und die Benutzeroberfläche mit Hilfe eines Benutzeroberflächen-Frameworks wie Java Swing zu gestalten – die Software nicht für verschiedene Prozessor- und Betriebssystemarchitekturen kompiliert werden muss.

Abbildung 32 zeigt die Startansicht der Benutzeroberfläche.

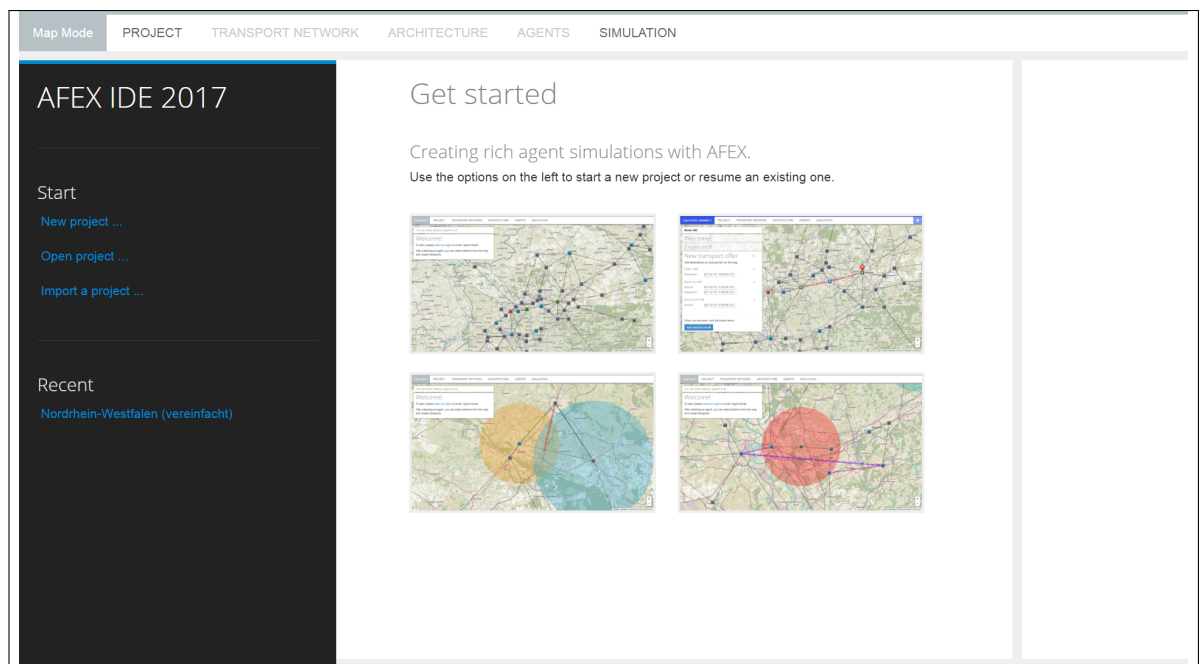


Abbildung 32: Startansicht der Benutzeroberfläche¹³⁰²

Unabhängig davon, dass die Benutzeroberfläche webbasiert ist, lässt sich die Entwicklungsumgebung auf einem handelsüblichen PC installieren und lokal betreiben. Es wäre jedoch ebenfalls möglich, die Software auf einem Server mit Internetzugang zu installieren und so mobil nutzbar zu machen. Die Entscheidung für die erst- oder zweitgenannte Art des Betriebs ist lediglich von den Nutzungspräferenzen der betreibenden Person abhängig und nicht durch technische Kriterien beeinflusst.

Die IDE ermöglicht durch die Benutzeroberfläche die Visualisierung des Verkehrsnetzes und der Präferenzen der Agentenkonfigurationen in einer Karte, wie die Benutzer sie von moderner Navigationssoftware kennen.

Wie bereits beschrieben, soll die prototypisch implementierte Software die weitere betriebliche sowie wissenschaftliche Anforderungsanalyse im Bereich Online-Frachtenbörsen unterstützen. Durch diese Fokussierung erhält der Aspekt „Benutzerfreundlichkeit“ eine erhöhte Priorität. Die grafische Benutzeroberfläche ermöglicht einem menschlichen Benutzer die vollständige und interaktive Erstellung und Verwaltung von Projekten und aller zugehörigen Objekte (insbesondere Szenarien) sowie die Durchführung und Auswertung von Simulationen. Aufgrund des bewusst hohen Abstraktionsgrades der Benutzeroberfläche sind zur Wahrnehmung dieser Aufgaben keine Kenntnisse in der Programmierung oder Entwicklung von Multi-Agenten-Systemen erforderlich. Es ist zu erwarten, dass ein

¹³⁰² Quelle: eigene Darstellung.

erhöhtes Maß an Benutzerfreundlichkeit die Akzeptanz für die dem Softwareprototyp zugrundeliegenden Ideen und Konzepte positiv beeinflussen wird.¹³⁰³

7.3.3.2 Persistenz

Die in der vorliegenden Forschungsarbeit prototypisch implementierten Agenten sind in der Lage, die in Kapitel 6 beschriebenen Aufgaben zu erfüllen.¹³⁰⁴

Zum Zwecke der Wiederverwendung von Agenten mit dem Ziel der wiederholten Durchführung gleicher Simulationen ist es jedoch nötig, alle Aspekte der Versuchsumgebung (eingegebene Daten sowie logische Verknüpfungen) über eine Sitzung und auch über einen Simulationsabbruch hinaus bereitzustellen.

Da die Agenten aufgrund ihrer Konzeption mit jedem Start neu initialisiert werden, muss die Persistierung der Versuchsumgebung außerhalb der Agentensoftware durch die IDE sichergestellt werden. Erreicht wird dies durch die Speicherung aller im Folgenden beschriebenen Objekte inklusive ihrer beschriebenen Verknüpfungen und Eigenschaften in einer Datenbank.

7.3.3.3 Projekte

Ein Projekt stellt eine abgeschlossene, eigenständige Struktur aus Verkehrsnetz, Agentenkonfigurationen, Szenarien, Simulationen und Simulationsergebnissen dar. Projekte dienen damit innerhalb der IDE als Container für individuelle, benutzergefertigte Testfälle.

Jedes Projekt kann aus einer Vorlage erstellt werden. Die individuellen Eigenschaften von Projekten sind:¹³⁰⁵

- Name,
- Beschreibung sowie
- Ausgangspunkt auf der Karte beim Laden des Projekts.

¹³⁰³ Dies begründet sich in folgender Plausibilitätsüberlegung: Studien zur Steigerung der Kaufbereitschaft und Konsumententreue legen nahe, dass eine erhöhte Benutzerfreundlichkeit die Bereitschaft einen Kauf zu tätigen und die Treue des Nutzers in Form wiederholter Käufe beim jeweiligen Online-Shop erhöht. Es erscheint plausibel, dass eine erhöhte Benutzerfreundlichkeit im Falle einer zur Anforderungsanalyse implementierten Software analog die Bereitschaft der Benutzer erhöht, sich mit den der Software zugrundeliegenden Ideen und Konzepten auseinanderzusetzen. Vgl. ZHANG et al. (2011), sowie CASALÓ et al. (2008).

¹³⁰⁴ Vgl. Kapitel 6, S. 115.

¹³⁰⁵ Eine technischere Beschreibung dieser Eigenschaften findet sich in Tabelle 24 im Anhang. Vgl. Anhang, S. 274.

Der Name sowie die Beschreibung dienen der Identifizierbarkeit des Projekts für den Benutzer.

Aus Gründen erhöhter Benutzerfreundlichkeit kann der Benutzer zudem für jedes Projekt festlegen, welcher Ausschnitt der Karte beim Laden des Projekts angezeigt werden soll.

Abbildung 33 zeigt beispielhaft die Eigenschaften eines Projekts.

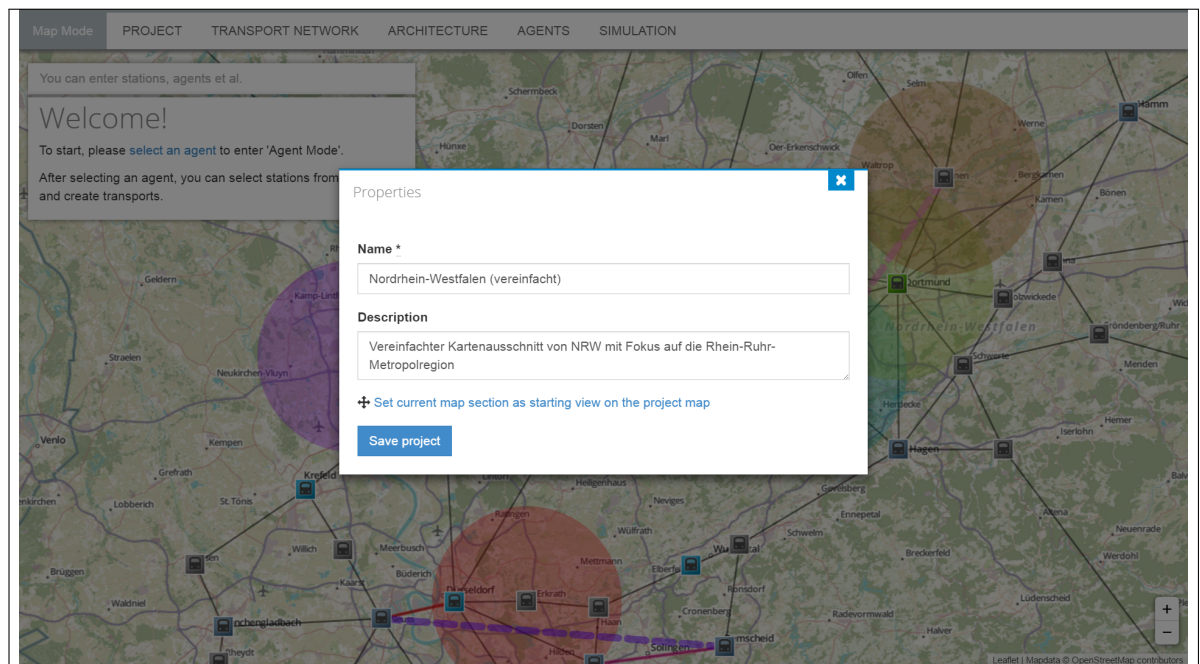


Abbildung 33: Beispielhafte Ansicht zu Projekteigenschaften¹³⁰⁶

7.3.3.4 Verkehrsnetz

Das Verkehrsnetz besteht, wie bereits beschrieben, aus Haltepunkten und Relationen.

Wird ein Projekt aus einer Vorlage erstellt, so sind durch diese Vorlage bereits Haltepunkte und Relationen im Verkehrsnetz des neuen Projekts vorhanden. Diese vorgegebenen Haltepunkte und Relationen können jedoch durch den menschlichen Benutzer nachträglich sowie projekt-individuell modifiziert und erweitert werden. Neue Projekte können zudem ohne Vorlage erstellt werden, so dass der Benutzer sämtliche Haltepunkte und Relationen manuell festlegen kann.

¹³⁰⁶ Quelle: eigene Darstellung.

Karten-Modus

Um innerhalb der IDE Haltepunkte zu erstellen und zu verwalten sowie Relationen zwischen ihnen zu definieren, bietet die IDE einen „Karten-Modus“.

Abbildung 34 zeigt beispielhaft die Kartenansicht eines Projekts im Karten-Modus.

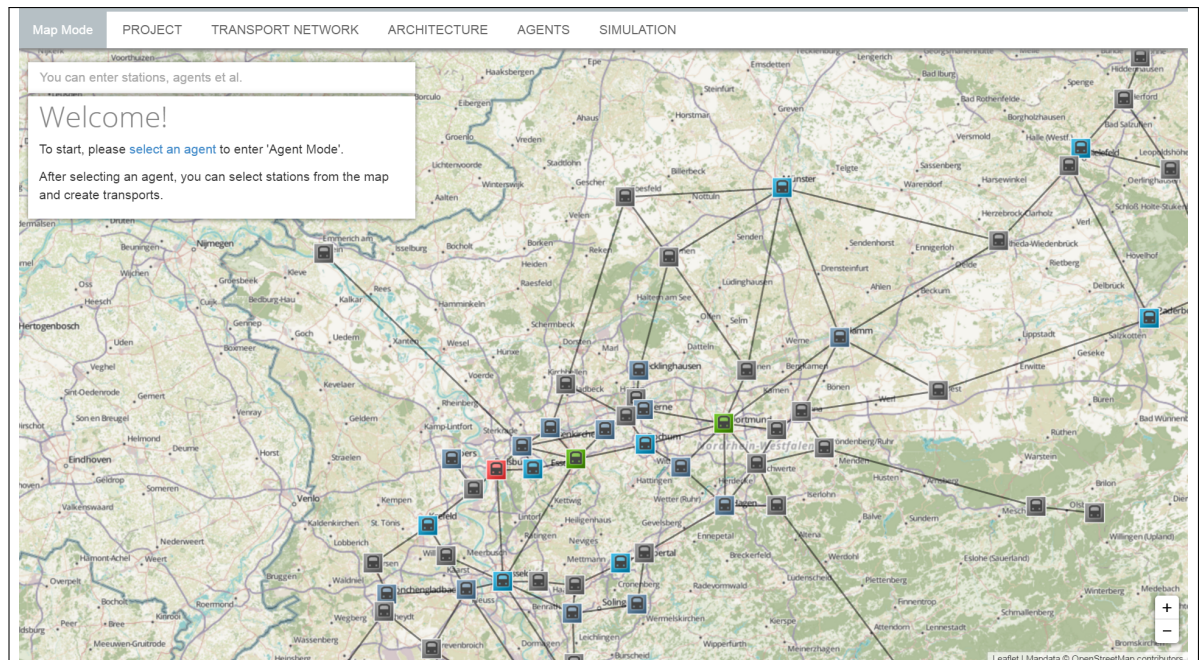


Abbildung 34: Beispielhafte Kartenansicht im Karten-Modus¹³⁰⁷

Feste Haltepunkte

Jeder feste Haltepunkt ist einem Projekt zugeordnet. Die individuellen Eigenschaften von Haltepunkten sind:¹³⁰⁸

- eindeutiger Schlüssel,
- Name,
- Breiten- und Längengrad sowie
- Priorität des Haltepunkts im Verkehrsnetz.

Zur digitalen Beschreibbarkeit der Haltepunkte muss jedem Haltepunkt ein einheitlicher und eindeutiger Schlüssel zugeordnet werden. Im Falle der verwendeten Bahnhofsdaten für Zughaltestellen wird dieser Schlüssel aus der internationalen Bahnstationsnummer (IBNR) gebildet.

Neben diesem eindeutigen Schlüssel wird jedem festen Haltepunkt zudem ein eindeutiger Name zugeordnet.

¹³⁰⁷ Quelle: eigene Darstellung.

¹³⁰⁸ Eine technischere Beschreibung dieser Eigenschaften findet sich in Tabelle 36 im Anhang. Vgl. Anhang, S. 280.

Der Haltepunkt wird mit Hilfe der Eigenschaften Breiten- und Längengrad auf der Karte verortet, welche als Gleitkommazahlen notiert werden.

Aus Gründen erhöhter Benutzerfreundlichkeit kann der Benutzer zudem für jeden Haltepunkt eine Priorität festlegen, welche die Darstellung des Haltepunkts in der Kartenansicht verändert. Haltepunkte mit niedriger Priorität werden erst eingeblendet, sobald der betrachtete Kartenausschnitt verkleinert wird, so dass in einem größeren Kartenausschnitt lediglich als „wichtig“ gekennzeichnete Haltepunkte einen Überblick über das Verkehrsnetz der betrachteten Region geben.

Abbildung 35 zeigt beispielhaft die vom Benutzer veränderbaren Eigenschaften eines festen Haltepunkts in der Kartenansicht.

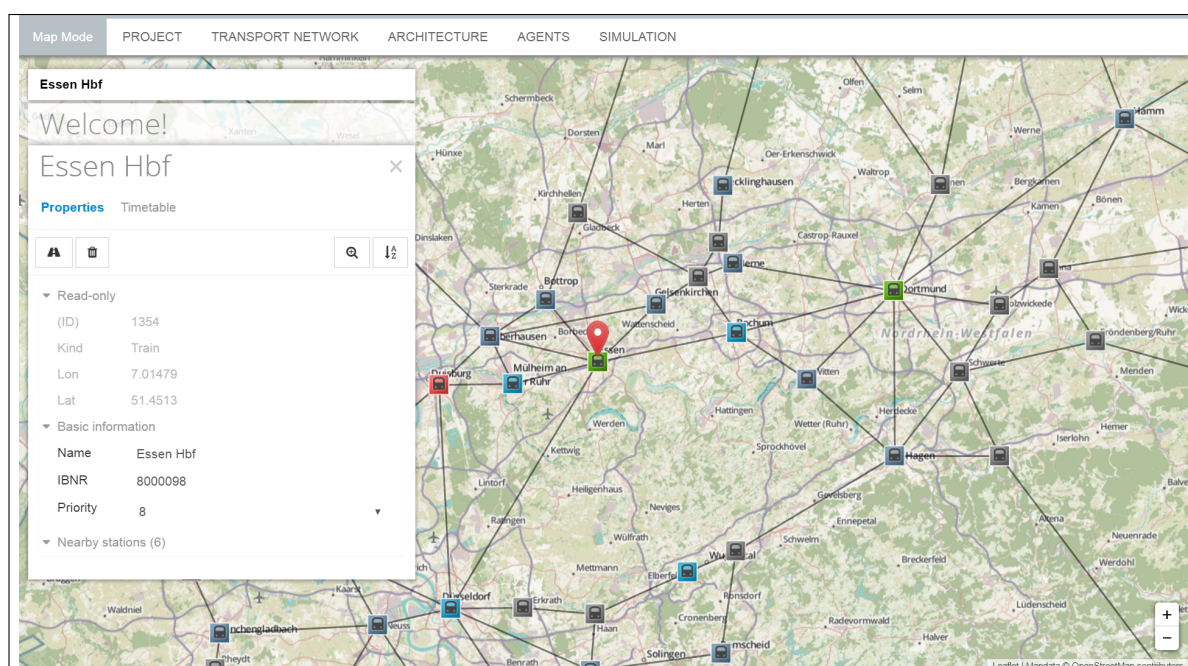


Abbildung 35: Beispielhafte Kartenansicht im Karten-Modus inklusive der Eigenschaften eines festen Haltepunkts¹³⁰⁹

Relationen

Jede Relation verbindet zwei feste Haltepunkte miteinander und ist hierdurch indirekt einem Projekt zugeordnet. Die individuellen Eigenschaften von Relationen sind:¹³¹⁰

- Startpunkt (fester Haltepunkt),
- Endpunkt (fester Haltepunkt) sowie
- Farbe zur Kenntlichmachung auf der Karte.

¹³⁰⁹ Quelle: eigene Darstellung.

¹³¹⁰ Eine technischere Beschreibung dieser Eigenschaften findet sich in Tabelle 35 im Anhang. Vgl. Anhang, S. 279.

Jede Relation wird durch ihren Start- und Endpunkt bestimmt.

Aus Gründen erhöhter Benutzerfreundlichkeit kann der Benutzer zudem für jede Relation die Farbe festlegen, mit der sie in der Kartenansicht eingezeichnet wird.

Datengrundlage zur Visualisierung des Verkehrsnetzes

Um das Verkehrsnetz in der IDE sowohl grafisch als auch schematisch abbilden zu können, wurde zunächst versucht, die relevanten europäischen Güterbahnhöfe sowie die zwischen ihnen bestehenden Streckenverbindungen zu ermitteln. Der Versuch wurde abgebrochen, obwohl Informationen zu Güterbahnhöfen öffentlich zugänglich sind und eine Reihe an Möglichkeiten identifiziert werden konnten, um Namen und Identifikationsnummern von europäischen Güterbahnhöfen zu sammeln: Die DB Cargo stellt Informationen zu Güterbahnhöfen, Güterfahrplänen und Ladestellen über das Internet bereit.¹³¹¹ Die HaCon GmbH betreibt für die DB Cargo eine internetbasierte Variante der Güterfahrpläne.¹³¹² Mit dem einheitlichen Entfernungszeiger für den internationalen Güterverkehr (DIUM) stehen Informationen über die Entfernungen zwischen Güterbahnhöfen bereit.¹³¹³

Die so gewonnenen Informationen wurden als nicht ausreichend erachtet, um ein möglichst realistisches Verkehrsnetz in der IDE darstellen zu können. Es fehlen Positionsangaben zu den einzelnen Bahnhöfen und die Information, zwischen welchen Bahnhöfen in der Realität Schienenstrecken verlaufen. Zusätzlich zu den ermittelten Güterbahnhöfen wurde auf die Daten des Projekts *OpenStreetMap* zurückgegriffen, um dem Benutzer eine vertraut wirkende Karte präsentieren zu können.

OpenStreetMap (OSM) ist ein 2004 am University College London initiiertes Projekt zur Sammlung aktueller geographischer Informationen, da diese Informationen in digitaler Form in europäischen Ländern nur unter Inkaufnahme hoher Kosten und in proprietären Formaten verfügbar waren.¹³¹⁴ Aufgrund der Höhe dieser Kosten waren Privatleute, kleine Unternehmen und öffentliche Vereine vielfach von ihrer Nutzung ausgeschlossen.¹³¹⁵ Aus diesem Grund stellt OpenStreetMap eine frei verfügbare, offene Sammlung geographischer Karteninformationen bereit, welche seit dem Start des Projekts von Freiwilligen erstellt und gepflegt wird.¹³¹⁶ Die Verfügbarkeit hochpräziser GPS-Informationen¹³¹⁷ über

¹³¹¹ Weitere Informationen sind im Internet verfügbar. Vgl. DB CARGO (2017).

¹³¹² Weitere Informationen sind im Internet verfügbar. Vgl. HACON (2017).

¹³¹³ Im Französischen „distancier international uniforme marchandises“. Weitere Informationen sind im Internet verfügbar. Vgl. DIUM (2017).

¹³¹⁴ Vgl. HAKLAY/WEBER (2008), S. 13.

¹³¹⁵ Vgl. HAKLAY/WEBER (2008), S. 13.

¹³¹⁶ Vgl. HAKLAY/WEBER (2008), S. 12.

¹³¹⁷ „GPS“ steht für „global positioning system“ und ist ein globales Navigationssatellitensystem zur Positionsbestimmung und Zeitmessung. Vgl. HAKLAY/WEBER (2008), S. 12.

kostengünstige GPS-Empfänger versetzt diese Freiwilligen in die Lage, Orte und Wege positionsgenau zu bestimmen und auf der öffentlichen Karte des OpenStreetMap-Projekts hinzuzufügen oder zu präzisieren.¹³¹⁸ Die Freiwilligen nehmen somit im Rahmen dieses „crowd-sourcing“ genannten Vorgangs Aufgaben wahr, welche von einzelnen Akteuren nicht leistbar wären und die sich derzeit noch nicht automatisieren lassen.¹³¹⁹ OpenStreetMap stellt somit ein ähnliches Unterfangen zur Generierung einer offenen Kartengrundlage dar, wie Wikipedia im Bereich der internetbasierten Nachschlagewerke.¹³²⁰ Es zielt darauf ab, einen Satz geographischer Daten bereitzustellen, welcher frei verfügbar und editierbar ist.¹³²¹

Abbildung 36 zeigt die Webseite des Projekts.¹³²²

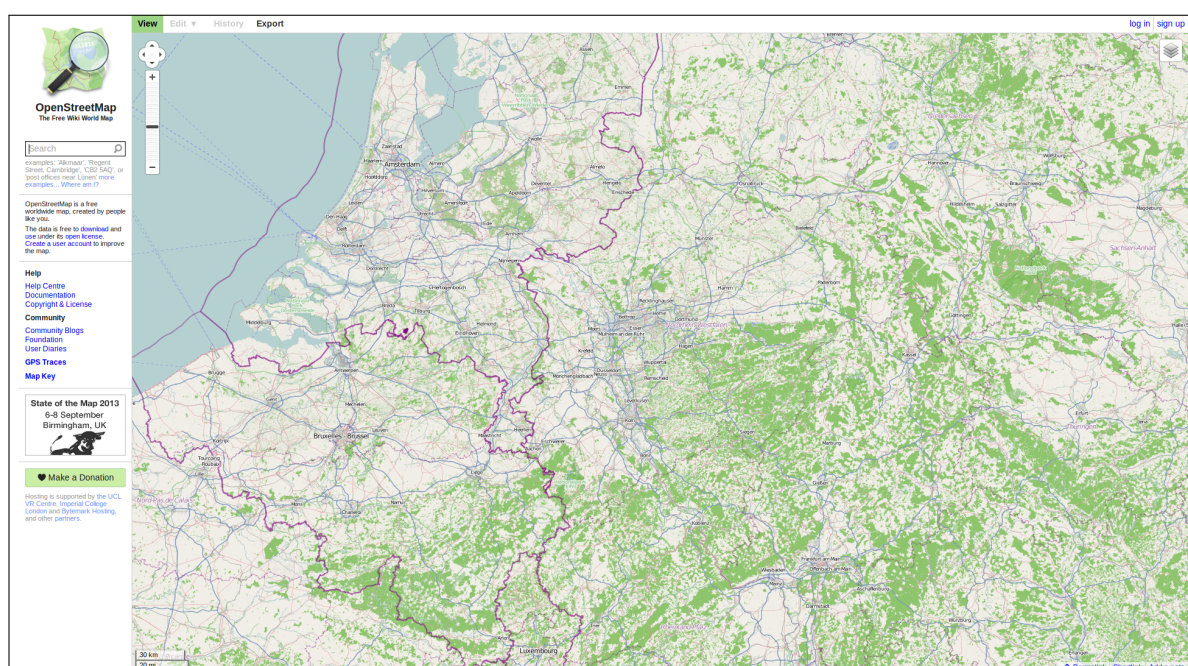


Abbildung 36: Webseite des OpenStreetMap-Projekts

Die Webseite präsentiert sich als Online-Kartendienst, ähnlich den Angeboten von Google¹³²³ und Microsoft¹³²⁴, welcher dem Benutzer das Bewegen, Hereinzoomen und Durchsuchen des angezeigten Kartenausschnitts erlaubt.¹³²⁵ Im Gegensatz zu den genannten kommerziellen Alternativen lässt sich bei OpenStreetMap jedoch der angezeigte Karten-

¹³¹⁸ Vgl. HAKLAY/WEBER (2008), S. 12.

¹³¹⁹ Vgl. HAKLAY/WEBER (2008), S. 12 f.

¹³²⁰ Vgl. HAKLAY/WEBER (2008), S. 13.

¹³²¹ Vgl. HAKLAY/WEBER (2008), S. 13.

¹³²² Weitere Informationen sind im Internet verfügbar. Vgl. OPENSTREETMAP (2017).

¹³²³ Weitere Informationen sind im Internet verfügbar. Vgl. GOOGLE MAPS (2017).

¹³²⁴ Weitere Informationen sind im Internet verfügbar. Vgl. BING MAPS (2017).

¹³²⁵ Vgl. HAKLAY/WEBER (2008), S. 13.

ausschnitt (inklusive aller dargestellten geographischen Daten) mittels einer Exportfunktionalität in verschiedenen offenen Formaten herunterladen.¹³²⁶

Eine „Bearbeiten“-Funktionalität erlaubt es Benutzern Kartendaten beizutragen.¹³²⁷

Die Daten des OpenStreetMap-Projekts werden immer häufiger zum Werkzeug und Gegenstand von Forschung.¹³²⁸ Allein in Deutschland konnten 30 Universitäten identifiziert werden, an denen Forschung mit Hilfe von OpenStreetMap-Daten betrieben wird oder diese Daten und die gesellschaftliche Relevanz von OpenStreetMap selbst Gegenstand andauernder Forschung sind.¹³²⁹

Die Qualität der von OpenStreetMap bereitgestellten Daten wurde in der Fachliteratur mehrfach untersucht¹³³⁰ und kritisiert, da die Daten teilweise unvollständig oder veraltet waren und nicht den tatsächlichen Gegebenheiten entsprachen. Jedoch gründet sich diese Kritik auf einen Vergleich des OpenStreetMap-Kartenmaterials mit dem Kartenmaterial kommerzieller Anbieter, wie bspw. Tele Atlas¹³³¹, und beschränkt sich auf spezielle Einsatzgebiete, wie bspw. die Nutzung des Kartenmaterials zur gezielten Navigation im Auto oder zu Fuß.¹³³² In städtischen Regionen wird eine höhere Datenqualität und -konsistenz festgestellt als in ländlichen Regionen.¹³³³

Die Datenqualität genügt den Anforderungen im Rahmen dieser Forschungsarbeit, da der Anspruch an die Datenqualität sich auf die korrekte Verortung von Bahnhöfen und Bahnstrecken innerhalb des Kartenmaterials beschränkt und genauere geographische Details, wie bspw. die genaue Positionierung von Gleisen, für die Abbildung des konzipierten Verkehrsnetzes als vernachlässigbar angesehen werden. Diese Feststellung folgt aus der Plausibilitätsüberlegung, dass die schematische Nachbildung von Gleisverbindungen zwischen Bahnhöfen geringere Anforderungen an die Datenqualität stellt, zu deren Erfüllung lediglich die Existenz einer Gleisverbindung zwischen zwei Bahnhöfen korrekt angegeben sein muss. Im Gegensatz zu Einsatzgebieten wie der Straßennavigation ist der tatsächliche Streckenverlauf nicht relevant.

¹³²⁶ Vgl. HAKLAY/WEBER (2008), S. 13 f.

¹³²⁷ Im Gegensatz zur Wikipedia haben sich die Initiatoren des OpenStreetMap-Projekts jedoch entschieden, nur registrierten Benutzern diese Funktionalität zu bieten, um so die Herkunft von Informationen, bspw. im Falle von Urheberrechtsfragen, feststellen zu können. Zudem kommen die Daten nicht ausschließlich von Privatpersonen. So hat Yahoo dem OpenStreetMap-Projekt im Jahr 2006 erlaubt, seine sonst urheberrechtlich geschützten Satellitenbilder zu benutzen, um mit ihnen den Verlauf von Straßen nachzuvollziehen. Im folgenden Jahr hat der niederländische Navigationsinformationsanbieter AND dem Projekt seine gesamte Straßenkarte der Niederlande gespendet. Auch die Regierung der „Isle of Man“ hat dem Projekt geographische Informationen über ihre Insel zukommenlassen.
Vgl. HAKLAY/WEBER (2008), S. 14.

¹³²⁸ Vgl. SCHMIDT/NEIS (2011), S. 27.

¹³²⁹ Vgl. SCHMIDT/NEIS (2011), S. 27.

¹³³⁰ Vgl. NEIS et al. (2011), HAKLAY (2010), sowie ZIELSTRA/ZIPF (2010).

¹³³¹ Vgl. ZIELSTRA/ZIPF (2010), S. 2.

¹³³² Vgl. SCHMIDT/NEIS (2011), S. 27.

¹³³³ Vgl. NEIS et al. (2011), S. 3 f., SCHMIDT/NEIS (2011), S. 27, sowie ZIELSTRA/ZIPF (2010), S. 14.

Abbildung 37 stellt diesen Sachverhalt schematisch dar.

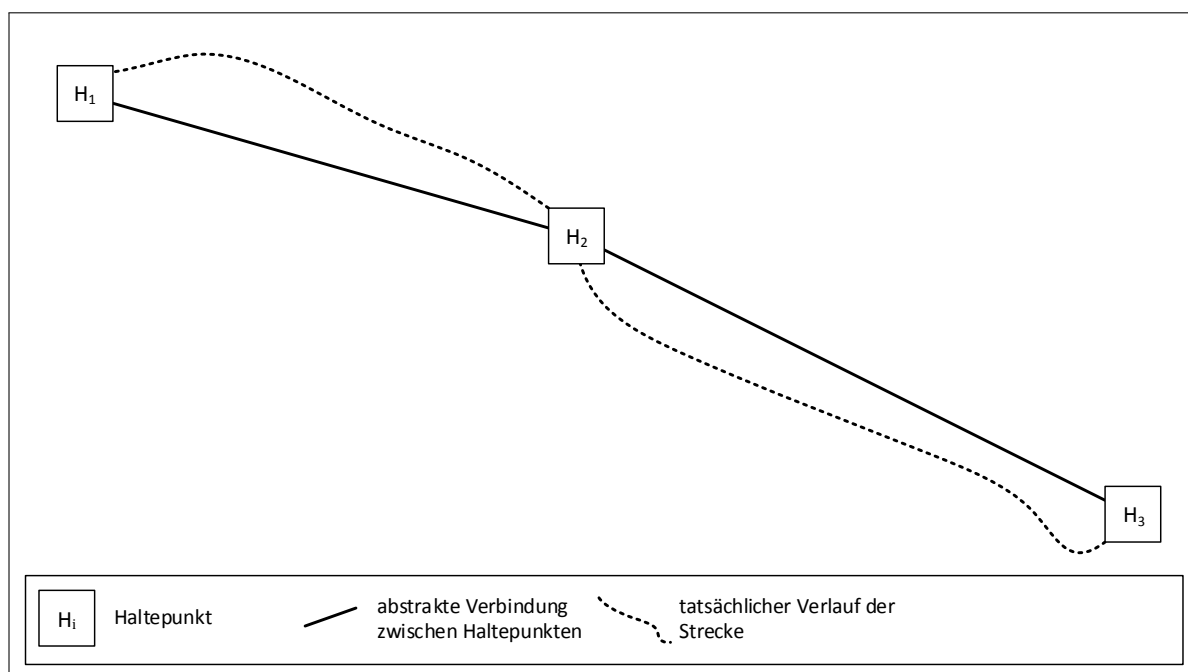


Abbildung 37: Abstrakte Verbindungen und tatsächliche Streckenverläufe¹³³⁴

Daher wurden für diese Forschungsarbeit die Kartendaten des „Korridors 24“ heruntergeladen und analysiert, um eine abstrakte, aber dennoch wirklichkeitsnahe Abbildung des Verkehrsnetzes innerhalb des CODE24-Gebiets zu ermöglichen.¹³³⁵

Es wurden ca. 5.700 Bahnhöfe im Bereich des Korridors 24 ermittelt, 2.879 davon in Deutschland. Eine verlässliche Unterscheidung, ob es sich bei diesen Bahnhöfen im Einzelfall um Betriebs-, Güter- oder Personenbahnhöfe (oder eine Kombination derselben) handelt, war auf Basis der verwendeten Datengrundlage nicht möglich.

Stattdessen wurde für die Ermittlung real existierender Gleisverbindungen auf die verfügbaren Informationen zum Öffentlichen Personennahverkehr zurückgegriffen.¹³³⁶ Anhand öffentlich verfügbarer Fahrpläne des Personennah- und -fernverkehrs konnten für den Zugverkehr relevante Verbindungen zwischen diesen Bahnhöfen ermittelt werden. Wie bereits ausgeführt und in Abbildung 37 dargestellt wurde, ist ein geringerer Detailgrad erforderlich als bei realen Anwendungen, da das Verkehrsnetz dieser Forschungsarbeit von realen Gegebenheiten abstrahiert. So wird es bspw. als ausreichend erachtet, wenn zur Beschreibung eines Gütertransports vom Hafen in Rotterdam an einen Gleisanschluss in Genua

¹³³⁴ Quelle: eigene Darstellung.

¹³³⁵ Die Daten des OpenStreetMap-Projekts stehen seit dem 2. September 2012 unter der Open Database Licence (ODbL), welche die Nutzung der bereitgestellten Daten regelt. Vgl. SCHULZE (2012), o.S. Weitere Informationen sind im Internet verfügbar. Vgl. OPENDATACOMMONS (2017).

¹³³⁶ Dies ist mit der Plausibilitätsüberlegung begründet, dass zur Demonstration des in dieser Forschungsarbeit entwickelten Verkehrsnetzes und Softwareprototyps eine vereinfachte Abbildung des real existierenden Schienennetzes ausreichend ist. Es wird angenommen, dass der öffentliche Personennahverkehr einen signifikanten Teil dieses Schienennetzes ebenfalls befährt und das so erhobene Schienenverkehrsnetz dem tatsächlichen Güterschienenverkehrsnetz nahekommmt.

von einem Transport im Hauptlauf auf der Schiene von „Rotterdam Centraal“ nach „Genova Piazza Principe“ ausgegangen wird, wenngleich dies nicht den realen Gegebenheiten entspricht.

Dieser bewusst in Kauf genommene geringere Detailgrad hat einen weiteren, praxisorientierten Grund: Während ein möglichst hoher Detailgrad bei einer späteren für den Produktiveinsatz implementierten Agenten-Software wünschenswert wäre, würde der rechnerische Aufwand, welchen die prototypisch implementierten Software-Agenten betreiben müssen, um eine Auktion bereits einfacher Transportdienstleistungen durchzuführen, erheblich steigen, wenn bereits relativ kurze Transporte über viele Haltepunkte laufen würden.

7.3.3.5 Agentenkonfigurationen

Die IDE ermöglicht es dem Benutzer, innerhalb eines Projekts mehrere Agentenkonfigurationen zu erstellen und zu verwalten. Eine Agentenkonfiguration beschreibt alle Daten, welche zum Starten und Betreiben einer Instanz der Agenten-Software nötig sind. Im Folgenden ist mit „Agentenkonfiguration“ stets dieser bestimmende, durch die IDE persistierte Datensatz und mit „Agent“ die mit Hilfe dieses Datensatzes gestartete Instanz der Agenten-Software gemeint.

Jede Agentenkonfiguration ist einem Projekt zugeordnet. Die individuellen Eigenschaften von Agentenkonfigurationen sind:¹³³⁷

- eindeutiger Schlüssel (UUID¹³³⁸),
- Name,
- Hostname oder IP-Adresse,
- Port sowie
- Farbe zur Kenntlichmachung auf der Karte.

Zur eindeutigen Identifizierbarkeit ist es notwendig, für jeden Agenten einen eindeutigen Schlüssel zu generieren, welcher den Agenten im Netzwerk unabhängig von seiner Adresse identifiziert. Es möglich, derartige über Zeit und Raum hinaus eindeutige Bezeichner zufällig zu generieren.¹³³⁹

Neben diesem technisch notwendigen, eindeutigen Bezeichner kann der Benutzer zudem einen nur in der IDE angezeigten, natürlichsprachlichen Namen vergeben. Dies dient der

¹³³⁷ Eine technischere Beschreibung dieser Eigenschaften findet sich in Tabelle 24 im Anhang. Vgl. Anhang, S. 274.

¹³³⁸ Im Englischen „universal unique identifier“ (UUID), teilweise auch „globally unique identifier“ (GUID). Vgl. LEACH et al. (2005).

¹³³⁹ Vgl. LEACH et al. (2005).

leichteren Identifizierung des Agenten in Simulationen durch den Benutzer und hat keinerlei Auswirkungen auf die Agentensoftware.

Hostname oder IP-Adresse sowie Port bilden die Adresse der Agenten im Internet.

Aus Gründen erhöhter Benutzerfreundlichkeit kann der Benutzer zudem für jede Agentenkonfiguration eine Farbe festlegen, mit welcher die angebotenen und nachgefragten Transporte des Agenten in der Karte eingezeichnet werden.

Jeder Agentenkonfiguration ist zudem ein Adressbuch zugeordnet, in welchem jene Adressen anderer Agenten verzeichnet sind, die dem Agenten bereits zu Anfang der Simulation bekannt sein sollen. Als Standardwert ist hier die Adresse des ersten im Projekt erstellten Agenten eingetragen.

Agenten-Modus

Die Kartenansicht dient nicht nur der Erstellung und Verwaltung von Haltepunkten, sondern auch der Erstellung und Verwaltung von Handelspräferenzen bestehender Agentenkonfigurationen.

Zur Verwaltung der Handelspräferenzen von Agentenkonfigurationen bietet die IDE neben dem Karten-Modus einen analogen „Agenten-Modus“.

Abbildung 38 zeigt diese Ansicht beispielhaft anhand der veränderten Ansicht eines festen Haltepunkts im Agenten-Modus im Vergleich zu Abbildung 35.

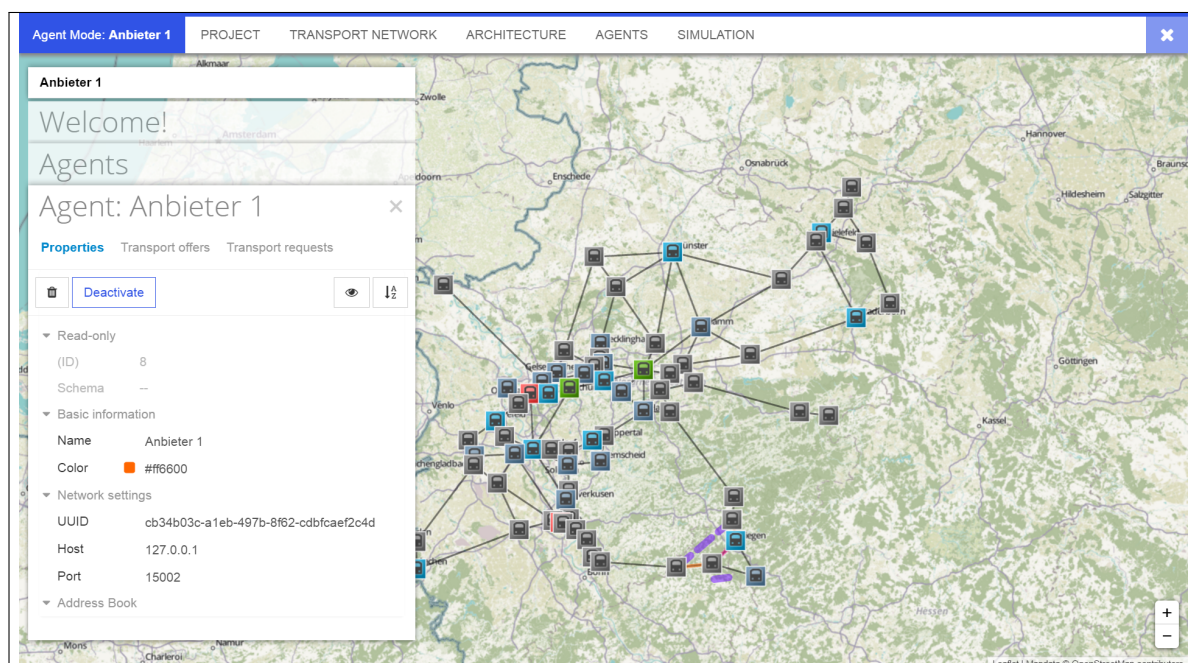


Abbildung 38: Beispielhafte Kartenansicht im Agenten-Modus inklusive Agenteneigenschaften¹³⁴⁰

¹³⁴⁰ Quelle: eigene Darstellung.

Jeder Agentenkonfiguration können Handelspräferenzen zugeordnet werden. Diese Handelspräferenzen stellen entweder ein Angebot für oder eine Nachfrage nach Transportdienstleistungen dar. Aus Gründen der Lesbarkeit werden die erstgenannten im Folgenden verkürzt „Angebotspräferenzen“ und die letztgenannten „Nachfragepräferenzen“ genannt.

Im Agenten-Modus präsentiert sich die Benutzeroberfläche in einer Art und Weise, wie die Benutzeroberfläche einer agenten-individuellen Client-Software aussehen könnte.¹³⁴¹ Der Benutzer übernimmt in diesem Modus die Rolle des von ihm ausgewählten Agenten. Wählt er nun bspw. feste Haltepunkte aus, so werden ihm andere Handlungsmöglichkeiten als im Karten-Modus geboten, wie bspw. das Erstellen einer neuen Angebots- oder Nachfragepräferenz von diesem Haltepunkt aus.

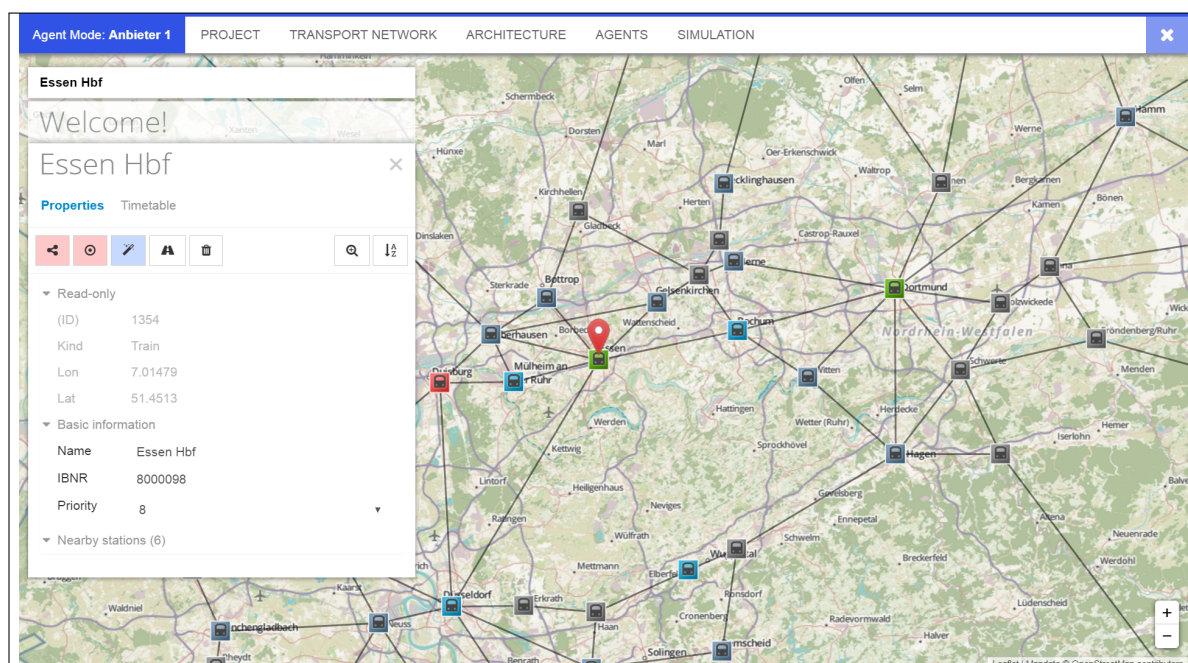


Abbildung 39: Beispielhafte Kartenansicht im Agenten-Modus inklusive der Eigenschaften eines festen Haltepunkts¹³⁴²

Zur digitalen Beschreibbarkeit der angebotenen bzw. nachgefragten Transportdienstleistung ist eine eindeutige Zuordnung der transportierbaren Güter (im Falle einer Angebotspräferenz) bzw. der zu transportierenden Güter (im Falle einer Nachfragepräferenz) notwendig.¹³⁴³ Das *Harmonisierte Güterverzeichnis* (NHM) ist ein internationales System zur einheitlichen Klassifizierung von Gütern im Eisenbahngüterverkehr, welches jedem Gut anhand von Güterbereich, Gütergruppe und Gutart einen eindeutigen, einheitlichen NHM-Code zuordnet.¹³⁴⁴

¹³⁴¹ Das Weiterentwicklungspotential im Bereich agenten-individueller Benutzeroberflächen bleibt hiervon unberührt. Vgl. Kapitel 10.2.3.2, S. 248.

¹³⁴² Quelle: eigene Darstellung.

¹³⁴³ Vgl. Kapitel 6.1, S. 115.

¹³⁴⁴ Vgl. UIC (2016).

Implementierung von Angebotspräferenzen

Innerhalb der IDE kann der Benutzer Angebotspräferenzen für feste und flexible Transporte definieren. Diese repräsentieren Transporte, die entweder ausschließlich aus festen oder flexiblen Haltepunkten bestehen.

Jede Angebotspräferenz ist einer Agentenkonfiguration und damit indirekt einem Projekt zugeordnet. Die individuellen Eigenschaften von Angebotspräferenzen sind:¹³⁴⁵

- Zeitraum, in welchem das Angebot gültig ist,
- eindeutige Güterkategorien (NHM¹³⁴⁶) derjenigen Güter, die transportiert werden können,
- maximal transportierbares Volumen in Kubikmeter,
- maximal transportierbares Gewicht in Kilogramm sowie
- minimaler Preis in EUR.

Der Gültigkeitszeitraum beschreibt, in welchem Zeitraum die Angebotspräferenz durch den Agenten kommuniziert werden soll. Der Agent wird die Angebotspräferenz weder vor Beginn noch nach Ende der Gültigkeit an andere Agenten kommunizieren. Beide Zeitpunkte, Beginn und Ende der Gültigkeit des Angebots, sind in koordinierter Weltzeit (UTC) notiert.¹³⁴⁷

Die transportierbaren Güter werden über ihre Güterkategorien (NHM) angegeben.

Das maximal transportierbare Volumen, das maximal transportierbare Gewicht sowie der nicht zu unterbietende Preis werden als Gleitkommazahlen angegeben. Wie schon bei der Notation von Zeitpunkten ist es auch für die Einheiten von Volumen, Gewichten und Währungen entscheidend, dass diese in einem einheitlichen Format notiert werden. Für die Zwecke dieser Forschungsarbeit wurden Kubikmeter, Kilogramm und EUR als Einheiten festgelegt, jedoch bieten die teils politisch und rechtlich geprägten Implikationen dieser Auswahl Potential für weitere Forschung.¹³⁴⁸

Zu den oben aufgeführten Eigenschaften besteht jede Angebotspräferenz zusätzlich aus Transportabschnitten, welche dem Angebot zugeordnet sind und folgende Eigenschaften besitzen:¹³⁴⁹

¹³⁴⁵ Eine technischere Beschreibung dieser Eigenschaften findet sich in Tabelle 28 im Anhang. Vgl. Anhang, S. 276.

¹³⁴⁶ Im Französischen „Nomenclature harmonisée des marchandises“. Vgl. UIC (2016).

¹³⁴⁷ UTC wurde gewählt, da es ein globaler externer Zeitstandard ist, der unabhängig von der Tatsache ist, in welchen Zeitzonen auf der Welt einzelne Agenten betrieben werden. Vgl. SCHMID (1995), S. 877.

¹³⁴⁸ Vgl. Kapitel 10.2.2.7, S. 247.

¹³⁴⁹ Eine technischere Beschreibung dieser Eigenschaften findet sich in Tabelle 27 im Anhang. Vgl. Anhang, S. 275.

- relative Position im Angebot,
- Startpunkt (fester Haltepunkt),
- frühester Abfahrtszeitpunkt am Startpunkt,
- Endpunkt (entweder als fester Haltepunkt oder als flexibler Haltepunkt in Form eines Radius um den Startpunkt, in welchem flexible Transporte möglich sind) sowie
- späterster Ankunftszeitpunkt am Endpunkt.

Die relative Position im Angebot bezeichnet einen null basierten Index, bei dem drei aufeinanderfolgenden Angebotsabschnitten die Positionen 0, 1 und 2 zugeordnet sind.

Start- und Endpunkt sind Referenzen auf feste Haltepunkte im Verkehrsnetz.

Der früheste Abfahrts- und der späteste Ankunftszeitpunkt sind in koordinierter Weltzeit (UTC) notierte Zeitpunkte.

Im Fall eines festen Transportangebots werden einer Angebotspräferenz beliebig viele Angebotsabschnitte zugeordnet.

Abbildung 40 zeigt den ersten Schritt zum Anlegen eines fixen Transports zwischen Essen und Dortmund: die Definition der am Transport beteiligten Haltepunkte.

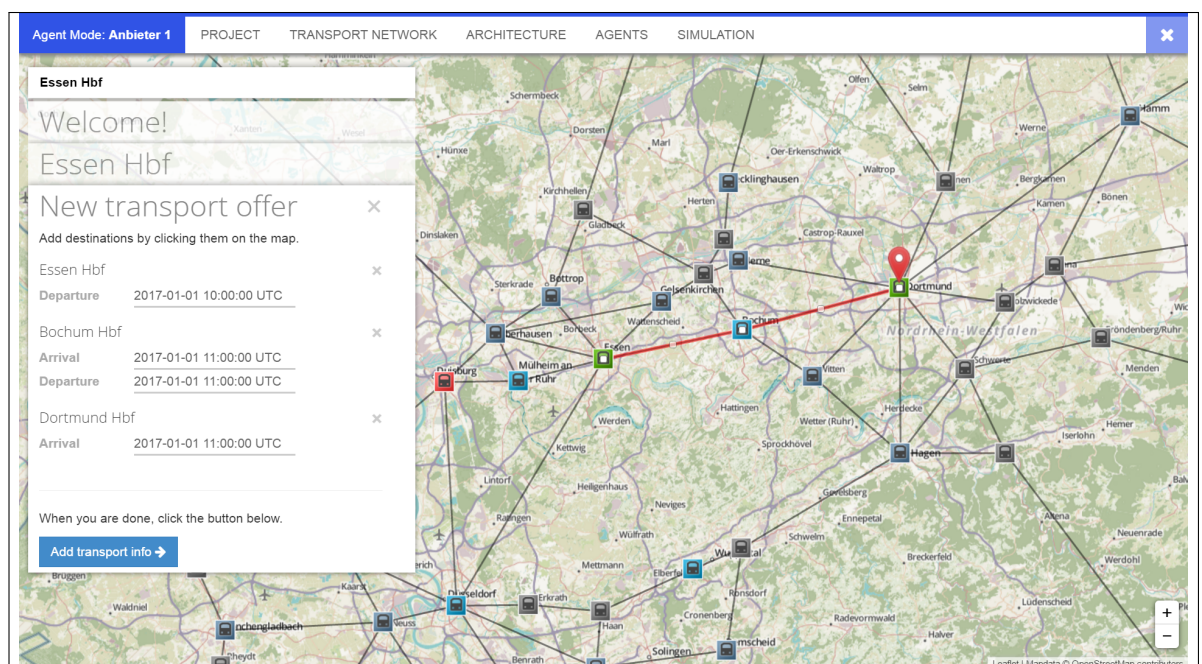


Abbildung 40: Anlegen einer Angebotspräferenz für einen fixen Transport im Agenten-Modus¹³⁵⁰

Im Fall eines flexiblen Transportangebots wird der Angebotspräferenz nur ein Angebotsabschnitt zugeordnet. Dieser Angebotsabschnitt besitzt keinen festen Endpunkt. Anstelle

¹³⁵⁰ Quelle: eigene Darstellung.

dessen besitzt er einen Radius, welcher als Ganzzahl notiert ist und der angibt, in welchem Umkreis um den Startpunkt der Anbieter Transporte anbieten möchte.

Analog zu Abbildung 40 zeigt Abbildung 41 den ersten Schritt zum Anlegen eines flexiblen Transports im Umkreis von 18 Kilometern um Essen.

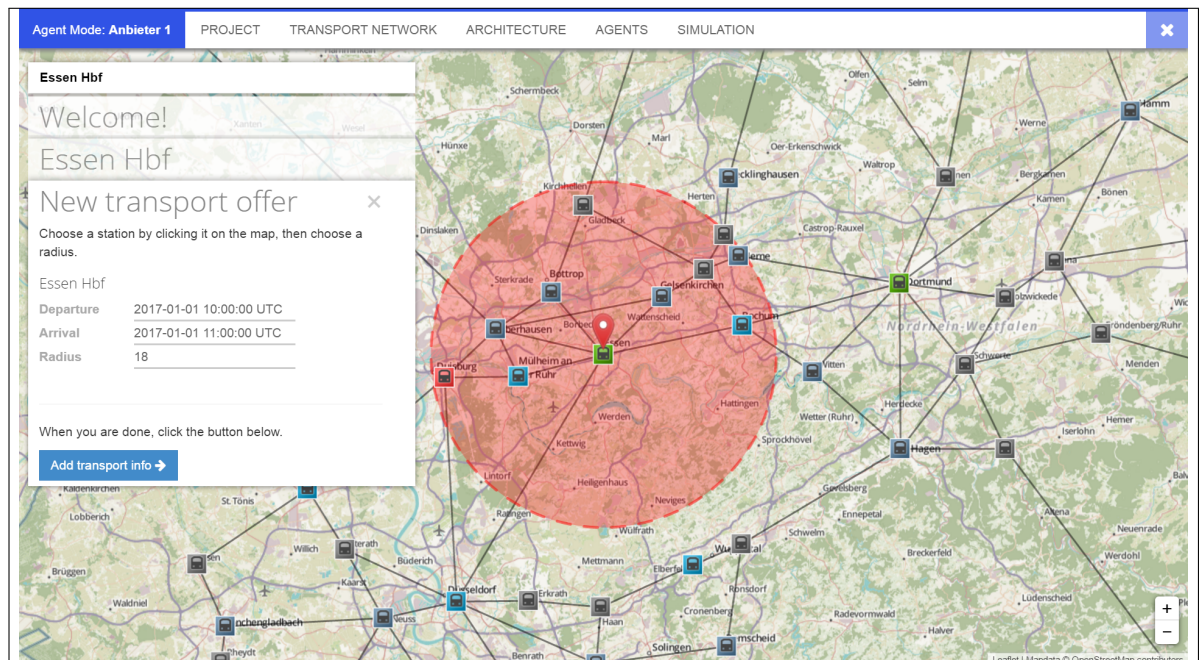


Abbildung 41: Anlegen einer Angebotspräferenz für einen flexiblen Transport im Agenten-Modus¹³⁵¹

Während sich sowohl feste als auch flexible Angebotspräferenzen stets an mindestens einem festen Haltepunkt orientieren, gilt diese Prämisse für Nachfragepräferenzen nicht.

Implementierung von Nachfragepräferenzen

Mit Nachfragepräferenzen können Benutzer innerhalb der IDE Nachfragen nach Transportdienstleistungen definieren. Sie unterscheiden sich strukturell von Angebotspräferenzen, da sie genau zwei Orte im Verkehrsnetz beschreiben, zwischen denen ein Transport nachgefragt wird. Die Orte („Wegpunkte“) können festen Haltepunkten oder beliebigen Postadressen entsprechen.

Abbildung 42 zeigt beispielhaft die Erstellung einer Nachfragepräferenz im Agenten-Modus.

¹³⁵¹ Quelle: eigene Darstellung.

Abbildung 42: Erstellung einer Nachfragepräferenz im Agenten-Modus¹³⁵²

Jede Nachfragepräferenz ist einer Agentenkonfiguration und damit indirekt einem Projekt zugeordnet. Die individuellen Eigenschaften von Nachfragepräferenzen sind:¹³⁵³

- Zeitraum, in welchem die Nachfrage gültig ist,
- eindeutige Güterkategorien (NHM) derjenigen Güter, die transportiert werden sollen,
- zu transportierendes Volumen in Kubikmeter,
- zu transportierendes Gewicht in Kilogramm,
- maximaler Preis in EUR,
- Startpunkt (Wegpunkt),
- frühester Abfahrtszeitpunkt am Startpunkt,
- Endpunkt (Wegpunkt) sowie
- spätester Ankunftszeitpunkt am Endpunkt.

Der Hauptteil dieser Eigenschaften ist analog zu den korrespondierenden Eigenschaften von Angebotspräferenzen definiert. Es ist jedoch zu beachten, dass sich die Semantik in den meisten Fällen unterscheidet.

¹³⁵² Quelle: eigene Darstellung.

¹³⁵³ Eine technischere Beschreibung dieser Eigenschaften findet sich in Tabelle 29 im Anhang. Vgl. Anhang, S. 277.

Der Gültigkeitszeitraum beschreibt, in welchem Zeitraum die Nachfragepräferenz durch den Agenten kommuniziert werden soll. Beginn und Ende der Gültigkeit der Nachfrage sind in koordinierter Weltzeit (UTC) notiert.

Die zu transportierenden Güter werden über ihre Güterkategorien (NHM) angegeben.

Das zu transportierende Volumen, das zu transportierende Gewicht sowie der (die maximale Zahlungsbereitschaft widerspiegelnde) maximale Preis werden als Gleitkommazahlen angegeben.

Im Gegensatz zu Angebotspräferenzen beinhalten Nachfragepräferenzen keine untergeordneten Abschnitte, sondern stets genau zwei Ortsangaben: Start- und Endpunkt der nachgefragten Transportdienstleistung, die im Fall von Nachfragepräferenzen als „Wegpunkte“ bezeichnet werden.

Der früheste Abfahrts- und späteste Ankunftszeitpunkt sind in koordinierter Weltzeit (UTC) notierte Zeitpunkte.

Wegpunkte von Nachfragepräferenzen

Wegpunkte sind entweder ein fester Haltepunkt, bspw. in Form eines Güterbahnhofs oder Seehafens, oder eine beliebige Postanschrift. Wegpunkte unterscheiden sich vom Konzept flexibler Haltepunkte durch ihre Spezifität: Sie bilden nicht eine Klasse von Punkten innerhalb eines Radius um einen festen Haltepunkt ab, sondern stellen genau einen Ort mit einer eindeutigen Anschrift und Position dar.

Die individuellen Eigenschaften von Wegpunkten sind:¹³⁵⁴

- fester Haltepunkt (optional),
- Breiten- und Längengrad,
- Straße,
- Hausnummer,
- Stadt,
- Region,
- Bundesland/-staat,
- Postleitzahl sowie
- Land.

¹³⁵⁴ Eine technischere Beschreibung dieser Eigenschaften findet sich in Tabelle 37 im Anhang. Vgl. Anhang, S. 280.

Ein Wegpunkt kann einem festen Haltepunkt zugewiesen werden. In diesem Fall ermittelt die IDE alle folgenden Eigenschaften anhand der Informationen des festen Haltepunkts. Der Grund für diese Option ist die erhöhte Benutzerfreundlichkeit/Zweckmäßigkeit.

Der Wegpunkt wird mit Hilfe der Eigenschaften Breiten- und Längengrad auf der Karte verortet, welche als Gleitkommazahlen notiert werden.

Alle weiteren Eigenschaften bilden die Postadresse des Wegpunktes und sind als alphanumerische Zeichenketten notiert, da auch Hausnummern und Postleitzahlen international Buchstaben enthalten können.

Durch die Agentenkonfigurationen mit ihren zugeordneten Angebots- und Nachfragepräferenzen bietet die IDE nun alle notwendigen Daten, welche nötig sind, um die Agenten zu starten und wie im vorherigen Kapitel beschrieben Auktionen durchführen zu lassen. Um die Durchführung dieser Simulationen klarer zu strukturieren bietet die IDE die Möglichkeit „Szenarien“ zu definieren.

7.3.3.6 Szenarien

Szenarien ermöglichen dem Benutzer der IDE, bei einer Simulation nur bestimmte Agentenkonfigurationen zu starten sowie die Agentenaktionen, Gruppenbildung sowie Auktionsergebnisse während der Auktion aufzuzeichnen und im Anschluss an diese auszuwerten.

So ist es durch die Verwendung von Szenarien leicht möglich, mehrere Szenarien zu erstellen, die sich nur geringfügig unterscheiden. Durch diese Vorgehensweise wird es möglich „vergleichende Experimente“ durchzuführen.

Zum Beispiel könnte der Benutzer zwei Szenarien erstellen und diese derart konfigurieren, dass an einem der beiden ein Agent mehr beteiligt ist. Dies würde dem Benutzer ermöglichen, nach der Simulation auszuwerten, wie sich die Gruppenbildung und das Auktionsergebnis verändern, wenn das Szenario bspw. einen weiteren Nachfrager enthält, der eine höhere Zahlungsbereitschaft hat, oder wenn ein weiterer Anbieter hinzukommt, der einen Transportabschnitt zu einem früheren Zeitpunkt anbietet.

Abbildung 43 zeigt Übersicht über alle Szenarien.

#	Name	Agents	Run Auctions	Previous Runs	
4	Spezialfall 4	Fall 4 - Anbieter 1 Fall 4 - Anbieter 2 Fall 4 - Anbieter 3 Fall 4 - Nachfrager 1	1	0	Prepare scenario
3	Spezialfall 3	Fall 3 - Anbieter 1 Fall 3 - Anbieter 2 Fall 3 - Anbieter 3 Fall 3 - Anbieter 4 Fall 3 - Nachfrager 1	1	0	Prepare scenario
2	Spezialfall 2	Fall 2 - Anbieter 1 Fall 2 - Anbieter 2 Fall 2 - Anbieter 3 Fall 2 - Nachfrager 1	1	0	Prepare scenario
1	Spezialfall 1	Fall 1 - Anbieter 1 Fall 1 - Anbieter 2 Fall 1 - Nachfrager 1 Fall 1 - Nachfrager 2	1	0	Prepare scenario

Abbildung 43: Übersicht über Szenarien¹³⁵⁵

Jedes Szenario ist einem Projekt zugeordnet.

Die individuellen Eigenschaften von Szenarien sind:¹³⁵⁶

- Name,
- zu startende Agentenkonfigurationen sowie
- Abbruchbedingung der Simulation.

Der Name dient der Identifizierbarkeit des Szenarios für den Benutzer.

Die zu startenden Agentenkonfigurationen sind eine Liste aller Agentenkonfigurationen, welche bei der Durchführung einer Simulation gestartet und überwacht werden sollen.

Die Abbruchbedingung gibt an, ob die Simulation so lange ausgeführt wird, bis der Benutzer sie manuell beendet, oder ob sie nach Durchführung einer bestimmten Anzahl erfolgreicher Auktionen automatisch beendet werden soll.

Für ein erstelltes Szenario müssen vor der Simulation spezifische Startparameter generiert werden. Da die IDE komplett unabhängig von der „eigentlichen“ Agenten-Software funktioniert, müssen die mit Hilfe der grafischen Benutzeroberfläche festgelegten Informationen in ein Format überführt werden, welches die prototypische Agenten-Software versteht. Die Startparameter umfassen die zum Starten der Agenten nötigen Startbefehle und Konfigurationsparameter.

Abbildung 44 zeigt diese Vorbereitung eines Szenarios beispielhaft.

¹³⁵⁵ Quelle: eigene Darstellung.

¹³⁵⁶ Eine technischere Beschreibung dieser Eigenschaften findet sich in Tabelle 34 im Anhang. Vgl. Anhang, S. 279.

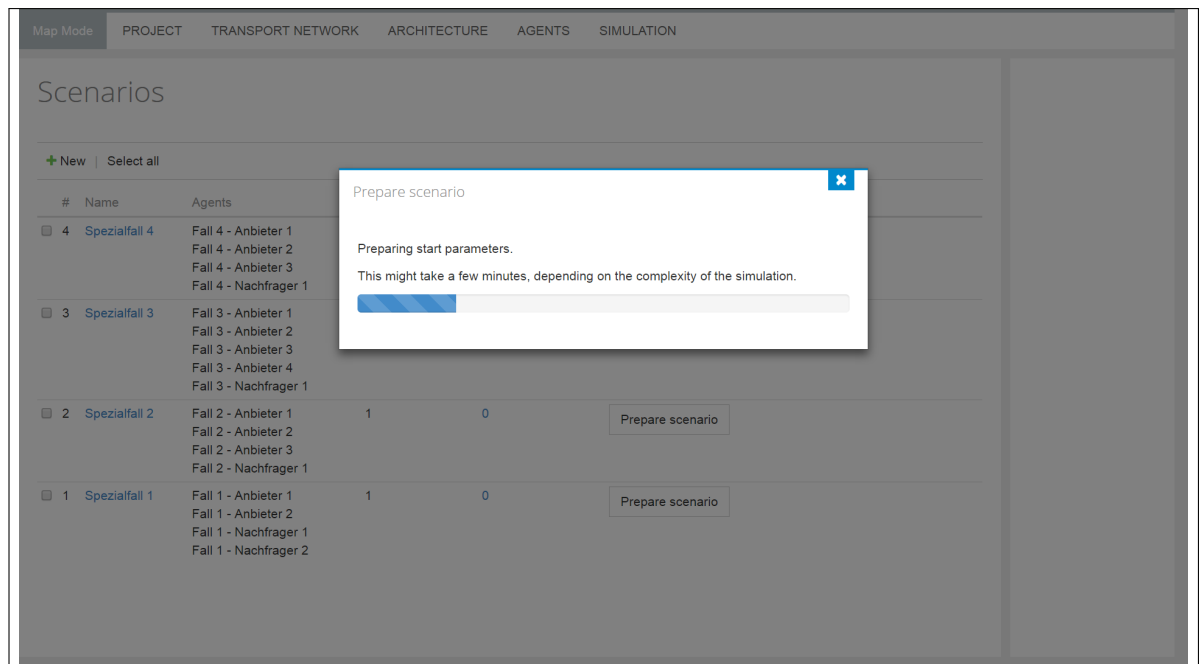


Abbildung 44: Beispielhafte Vorbereitung eines Szenarios zur Simulation¹³⁵⁷

Nach der Vorbereitung des Szenarios kann dieses sofort als Simulation gestartet werden. Werden die zugrunde liegenden Daten wie Haltepunkte, Agentenkonfigurationen und Handelspräferenzen nicht verändert, so kann die Simulation des Szenarios beliebig oft wiederholt werden, ohne die Startparameter erneut generieren zu müssen.

7.3.3.7 Simulationen

Bei jeder Simulation startet die IDE die im angegebenen Szenario aufgeführten Agenten mit Hilfe der zuvor generierten Startparameter. Der Begriff „Simulation“ ist hierbei auf die Startsituation bezogen, nicht jedoch auf die technischen Bedingungen, unter denen die Agenten-Software betrieben wird. Während der Simulation werden reguläre Instanzen der prototypisch implementierten Agenten-Software gestartet, welche über ihre regulären Netzwerkprotokolle, sowie die agenten-spezifischen Sprachen und Nachrichten miteinander kommunizieren. Der Begriff „Simulation“ wird verwendet, da diese authentischen Instanzen der Agenten-Software sowohl in ihrem Start als auch ihrer Beendigung durch die IDE orchestriert werden. Die einzige Modifikation der Kommunikationsprozesse der Agenten besteht darin, dass alle Agenten zusätzlich zu ihren normalen Routinen Signale über ihr Handeln an die IDE senden.

Während der Durchführung werden alle gestarteten Agenten mittels dieser Signale überwacht und ihre Aktionen für die spätere Auswertung protokolliert.

¹³⁵⁷ Quelle: eigene Darstellung.

Durchführung von Simulationen

Die Benutzeroberfläche der IDE gibt in der Simulationsansicht grafisch wieder, welche Agenten gestartet sind und welche Gruppen sich gebildet haben.

Abbildung 45 zeigt diese Ansicht beispielhaft.

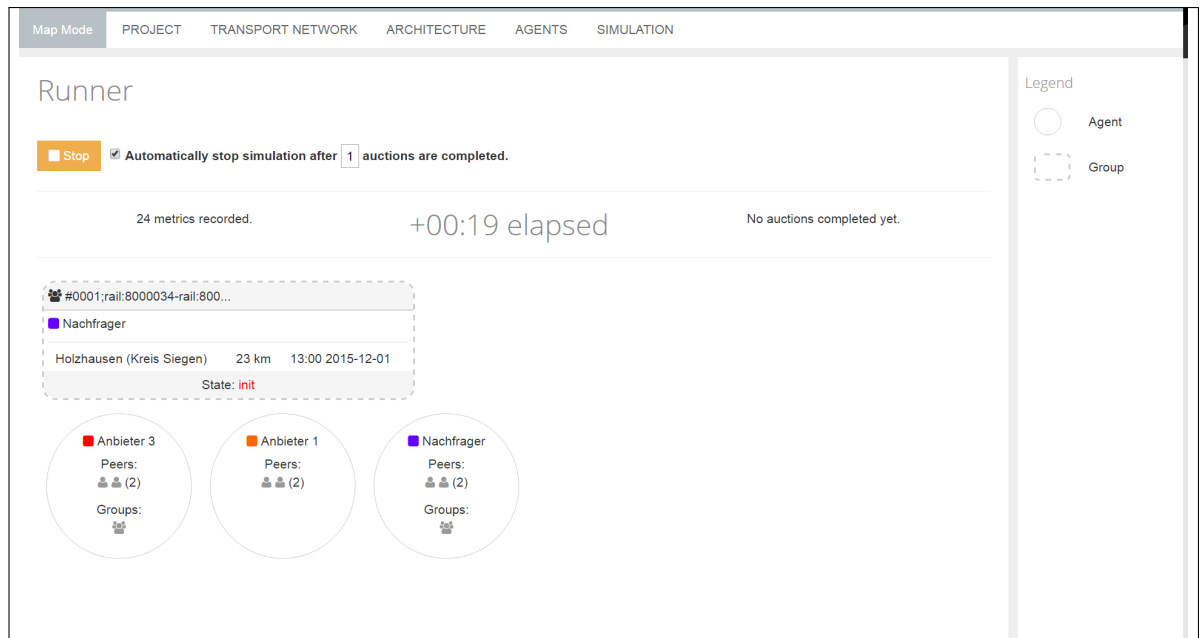


Abbildung 45: Startansicht einer Simulation¹³⁵⁸

Die Ansicht unterteilt sich in zwei Abschnitte: Agenten und Gruppen.

Im Abschnitt „Agenten“ tauchen nacheinander alle Agenten auf, sobald sie gestartet wurden (als Kreise dargestellt). Für jeden Agenten wird dargestellt, wie viele andere Agenten er bereits im dezentralen Netzwerk gefunden hat und an wie vielen Gruppen er beteiligt ist.

Im Abschnitt „Gruppen“ tauchen alle von den Agenten im Verlauf der Simulation gegründeten, zum jeweiligen Zeitpunkt aktiven Gruppen auf (als Rechtecke dargestellt). Nicht angezeigt werden finale und zusammengeführte Gruppen.

Für jede Gruppe wird aufgeführt, welche Agenten beteiligt sind, welche Güter innerhalb der Gruppe versteigert werden und in welchem Zustand bzw. welcher Phase sich die Gruppe befindet.

Der Benutzer hat unabhängig von der im laufenden Szenario definierten Abbruchbedingung jederzeit die Möglichkeit, die Simulation manuell zu beenden. Ist eine definierte Abbruchbedingung erfüllt, endet die Simulation automatisch.

¹³⁵⁸ Quelle: eigene Darstellung.

Auswertung von Simulationen

Nach Ende der Simulation wird eine Zusammenfassung der Ergebnisse angezeigt.

Abbildung 46 zeigt diese Zusammenfassung beispielhaft.

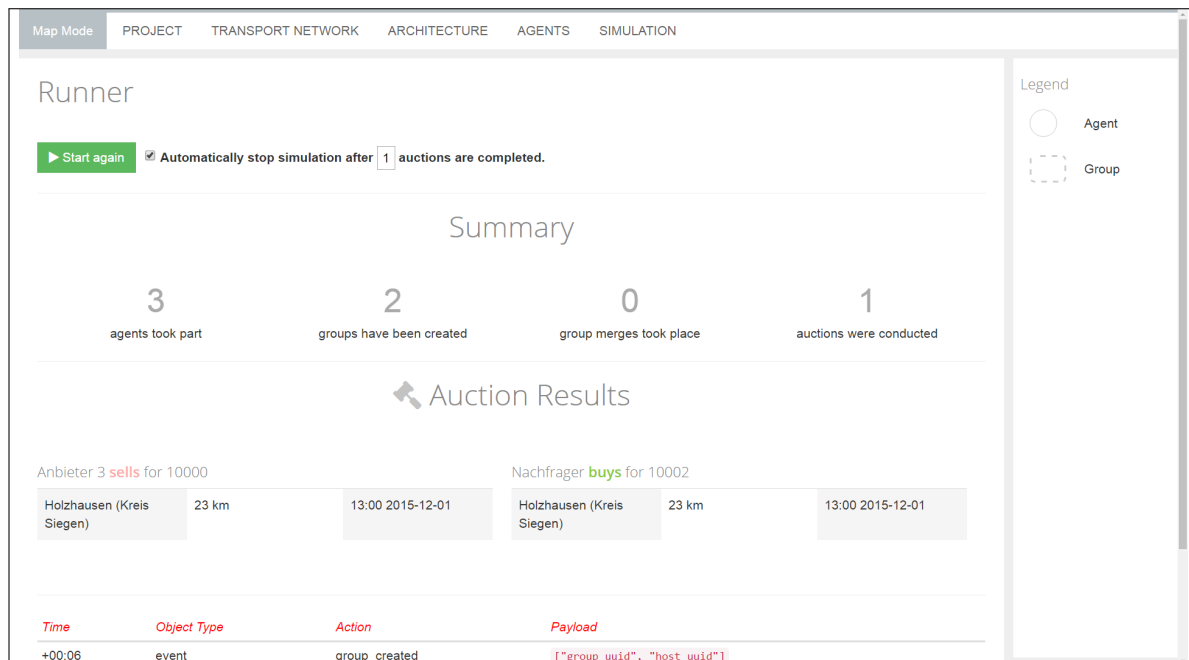


Abbildung 46: Zusammenfassung der Ergebnisse einer Simulation¹³⁵⁹

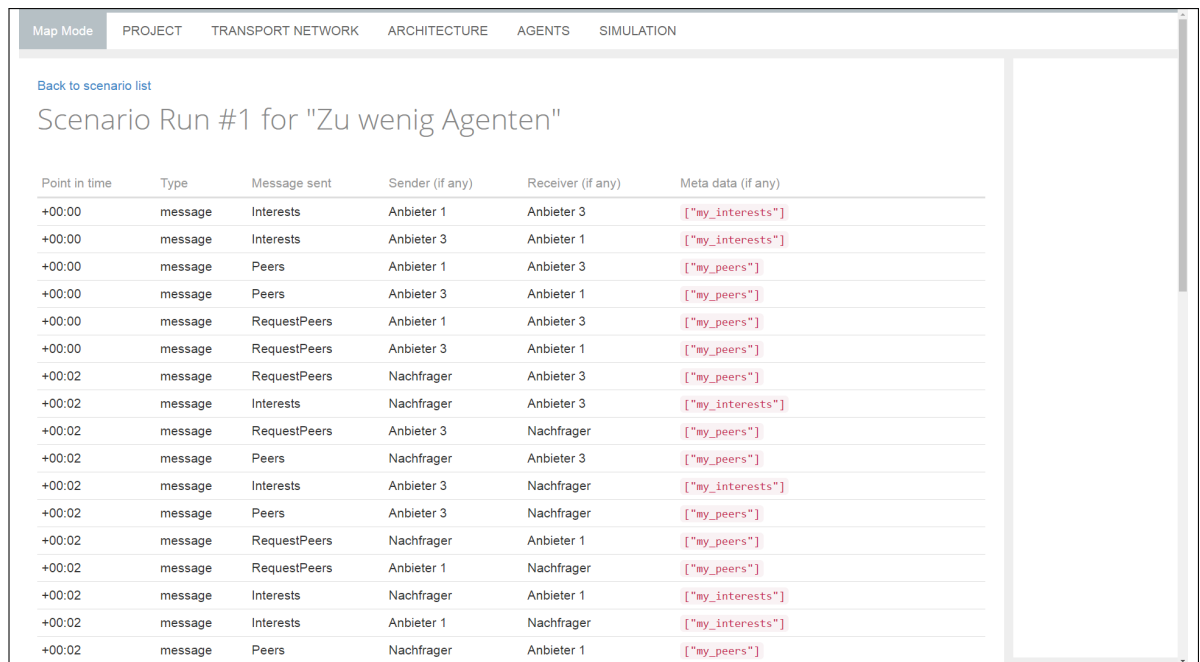
In der Zusammenfassung werden die folgenden Daten aufgelistet:

- Anzahl der gestarteten Agenten,
- Anzahl der von gestarteten Agenten gegründeten Gruppen,
- Anzahl der im Laufe der Simulation zusammengeführten Gruppen,
- Anzahl der durchgeführten Auktionen sowie
- Informationen darüber, welche Transportdienstleistungen im Rahmen der durchgeführten Auktionen von welchem Agenten verkauft oder gekauft wurden.

Neben dieser Zusammenfassung steht dem Benutzer auch das detaillierte Protokoll aller zwischen den Agenten gesendeten Nachrichten zur Ansicht bereit.

Abbildung 47 zeigt diese Ansicht beispielhaft.

¹³⁵⁹ Quelle: eigene Darstellung.



Point in time	Type	Message sent	Sender (if any)	Receiver (if any)	Meta data (if any)
+00:00	message	Interests	Anbieter 1	Anbieter 3	["my_interests"]
+00:00	message	Interests	Anbieter 3	Anbieter 1	["my_interests"]
+00:00	message	Peers	Anbieter 1	Anbieter 3	["my_peers"]
+00:00	message	Peers	Anbieter 3	Anbieter 1	["my_peers"]
+00:00	message	RequestPeers	Anbieter 1	Anbieter 3	["my_peers"]
+00:00	message	RequestPeers	Anbieter 3	Anbieter 1	["my_peers"]
+00:02	message	RequestPeers	Nachfrager	Anbieter 3	["my_peers"]
+00:02	message	Interests	Nachfrager	Anbieter 3	["my_interests"]
+00:02	message	RequestPeers	Anbieter 3	Nachfrager	["my_peers"]
+00:02	message	Peers	Nachfrager	Anbieter 3	["my_peers"]
+00:02	message	Interests	Anbieter 3	Nachfrager	["my_interests"]
+00:02	message	Peers	Anbieter 3	Nachfrager	["my_peers"]
+00:02	message	RequestPeers	Nachfrager	Anbieter 1	["my_peers"]
+00:02	message	RequestPeers	Anbieter 1	Nachfrager	["my_peers"]
+00:02	message	Interests	Nachfrager	Anbieter 1	["my_interests"]
+00:02	message	Interests	Anbieter 1	Nachfrager	["my_interests"]
+00:02	message	Peers	Nachfrager	Anbieter 1	["my_peers"]

Abbildung 47: Detailliertes Protokoll einer Simulation¹³⁶⁰

In dieser Ansicht sind für jede Nachricht folgende Informationen aufgeführt:

- Zeitpunkt der Nachricht im Verlauf der Simulation relativ zu deren Beginn,
- Typ der Nachricht,
- Sender der Nachricht,
- Empfänger der Nachricht sowie
- in der Nachricht enthaltene Metadaten.

Eine Auflistung aller Nachrichtentypen inklusive der zugehörigen Metadaten findet sich im Anhang.¹³⁶¹

Im Rahmen der Durchführung von Simulationen muss die IDE die Agenten starten. Diese auf Basis aller zuvor genannter Daten gestarteten Agenten sind eine eigenständige Software: die Agenten-Software, deren separate Implementierung im folgenden Kapitel beschrieben wird.

¹³⁶⁰ Quelle: eigene Darstellung.

¹³⁶¹ Vgl. Anhang, S. 341.

7.4 Implementierung der Agenten-Software

7.4.1 Grundlagen

Die Kernaufgabe der Agenten-Software besteht in der Umsetzung von vorgegebenen Benutzerwünschen in Form von Präferenzen. Da die vorliegende prototypische Implementierung der Agenten-Software keine eigene grafische Benutzeroberfläche besitzt, müssen ihr die beim Start relevanten Parameter auf andere Art kommuniziert werden. In einer eventuell später stattfindenden, markttauglichen Implementierung werden diese Präferenzen über eine mitgelieferte Eingabemaske direkt innerhalb der Agenten-Software eingegeben. Für die Zwecke dieser Disseration wurde die Eingabe der Benutzerpräferenzen in die IDE verschoben, so dass es möglich ist, mehrere Agentenkonfigurationen parallel zu erstellen, zu verwalten und zu steuern, ohne hierfür mehrere Instanzen der Agenten-Software installieren und betreiben zu müssen.

Aus diesem Grund ist die prototypische Implementierung der Agenten-Software darauf ausgelegt, die zu ihrem Betrieb zwangsläufig notwendigen Präferenzen extern als Teil der Startparameter vorgegeben zu bekommen. Dies geschieht, indem die IDE nach dem Start einer Agenteninstanz die innerhalb ihrer Benutzeroberfläche definierten Präferenzen an die Agenteninstanz weiterleitet. Aus technischer Sicht macht es keinen Unterschied, ob die Präferenzen in der Software selbst oder über eine Schnittstelle eingegeben werden. Eine Erweiterung der Agenten-Software um eine solche Benutzerschnittstelle in Form einer Eingabemaske ist somit denkbar und mit geringem Aufwand zu erreichen.¹³⁶²

Analog zur Verwendung eines Webframeworks zur Erstellung der Benutzeroberfläche der integrierten Entwicklungsumgebung wird an dieser Stelle noch auf die unterlassene Verwendung eines „Agenten-Frameworks“ zur Erstellung der Agenten-Software eingegangen.¹³⁶³

Die Verwendung eines solchen Frameworks war für die Aufgabe der Implementierung des Softwareprototyps in dieser Forschungsarbeit kritisch zu hinterfragen. Einerseits versprechen Frameworks einen „Startvorteil“, da sie häufig verwendete Komponenten eines Softwaresystems abstrahiert zur Verfügung stellen, so dass diese nicht vom Entwickler selbst programmiert werden müssen. Beispiele hierfür sind die Anbindung an eine Datenbank und das Austauschen von Nachrichten zwischen Agenten.

Somit erscheint die Verwendung eines Frameworks zur Erstellung und Simulation eines Multi-Agenten-Systems den Zwecken dieser Forschungsarbeit dienlich. NIKOLAI und MADEY diskutieren 53 verschiedene Projekte, welche in verschiedener Weise Werkzeuge zur partiellen Erfüllung dieser Aufgabe zur Verfügung stellen und teilweise den Standards der

¹³⁶² Vgl. Kapitel 10.2.3.2, S. 248.

¹³⁶³ In der Literatur wird auch der Begriff „Toolkit“ verwendet. Für die Zwecke dieser Forschungsarbeit sind beide Begriffe als synonym anzusehen. Vgl. NIKOLAI/MADEY (2009).

FIPA¹³⁶⁴ folgen.¹³⁶⁵ Die identifizierten Projekte decken mehrere Programmiersprachen ab und es konnten sowohl Open- als auch Closed-Source-Projekte identifiziert werden.¹³⁶⁶ Die meisten Projekte unterstützen eine Reihe von Betriebssystemen.¹³⁶⁷ Doch Frameworks sind stets Werkzeuge, welche eine generalisierte Lösung für eine bestimmte Anzahl abstrakt betrachteter Probleme darstellen.

Analog zu den in Kapitel 7.2.3.2 diskutierten allgemeinen Programmiersprachen stellen Frameworks generische Ansätze der Problemlösung für weite und offene Problemdomänen dar, wie bspw. „Entwicklung eines internetbasierten Softwaresystems zur Verwaltung von Geschäftsdaten“.¹³⁶⁸ Generische Ansätze verlieren jedoch durch ihre höhere Allgemeingültigkeit ihre Effektivität bei der Problemlösung in einer enger definierten Problem-domäne wie bspw. „Implementierung eines prototypischen Systems autonomer, dezentral organisierter Software-Agenten für die Durchführung zweiseitiger kombinatorischer Auktionen“.

Der Aufwand einer partiellen Neuimplementierung der entscheidenden Aspekte eines Frameworks¹³⁶⁹ wurde vom Autor als nicht wesentlich niedriger eingeschätzt als der Rüstaufwand, den die Verwendung und Anpassung einer nicht für die Zwecke dieser Forschungsarbeit angefertigten Standardlösung mit sich gebracht hätten. Wegen dieser Indifferenz in Bezug auf den zeitlichen Aufwand ist das entscheidende Argument für die partielle Neuimplementierung das durch die eigenständige Implementierung erwartete tiefere Verständnis der Problemdomäne.

7.4.2 Schnittstelle der IDE zur Agenten-Software

7.4.2.1 Notwendigkeit von Startparametern

Da die vorliegende prototypische Implementierung der Agenten-Software keine grafische Benutzeroberfläche besitzt, müssen ihr die beim Start relevanten Parameter auf andere Art kommuniziert werden. Diese Startparameter unterteilen sich in Konfigurationsparameter

¹³⁶⁴ Die Foundation for Intelligent Physical Agents (FIPA) wurde 1996 mit dem Ziel gegründet, ein Standardisierungsgremium zur Schaffung von Richtlinien für Agentenkommunikation, Agentenmanagement und den Transport von Agentennachrichten zu schaffen.

Weitere Informationen sind im Internet verfügbar. Vgl. FIPA (2017).

¹³⁶⁵ Vgl. NIKOLAI/MADEY (2009).

¹³⁶⁶ Vgl. NIKOLAI/MADEY (2009).

¹³⁶⁷ Vgl. NIKOLAI/MADEY (2009).

¹³⁶⁸ Vgl. Kapitel 7.2.3.2, S. 169.

¹³⁶⁹ Es handelt sich nicht um eine vollständige Neuimplementierung, da viele mathematische, die Kommunikation sowie die Datenhaltung betreffende Module als vorgefertigte und eigenständig vertriebene Open-Source-Module verwendet werden können. Somit muss bspw. die Kommunikation zwischen Agenten in Bezug auf die agenten-spezifische Sprache selbst implementiert werden, nicht jedoch das Versenden und Empfangen der resultierenden Daten über das Internet.

und Handelspräferenzen, welche beim Start des Agenten geladen werden. Die Konfigurationsparameter sind der eindeutige Schlüssel des Agenten (UUID) sowie seine Adresse (IP oder Hostname und Port).

Damit die Agenten-Software in der Lage ist, die in der IDE beschriebenen Präferenzen für Transportdienstleistungen in Auktionsgebote umzusetzen, müssen die in der Datenbank der IDE vorliegenden Informationen in eine für die Agenten verständliche, einheitliche Form konvertiert werden. Dies führt dazu, dass zu jeder Agentenkonfiguration in der IDE genau ein Agent mit den zu der Konfiguration gehörigen Präferenzen gestartet werden kann.

Die Agenten sind so konzipiert, dass ihnen das ihrer Tätigkeit zugrunde liegende Verkehrsnetz nur durch die geäußerten Präferenzen der Agenten bekannt ist. Dies wird mit der Plausibilitätsüberlegung begründet, dass eine von der konkreten Struktur des Verkehrsnetzes unabhängige Auktionsmethode wünschenswert ist, da sie die Simulation möglichst vieler unterschiedlicher Szenarien mit abweichenden Verkehrsnetzen ermöglichen soll.

Die angebotenen Transportdienstleistungen stellen zu jedem Zeitpunkt einer Simulation das „ersteigerbare Verkehrsnetz“ dar. Die Kenntnis von möglichen, jedoch nicht angebotenen Transportstrecken würde für die nachfragenden Agenten keinerlei Mehrwert bieten. Die Vorgehensweise ist daher, die bei Beginn einer Simulation angebotenen Transportdienstleistungen in eine für die Simulation einheitliche Notation zu überführen. Alle nachfragenden Präferenzen werden anschließend ebenfalls in dieser Form notiert, so dass anbietende und nachfragende Agenten sich einer einheitlichen Sprache bedienen.¹³⁷⁰

7.4.2.2 Notation

Ein Angebot besteht aus mehreren Transportabschnitten. Jeder Transportabschnitt muss wie beschrieben durch die Schnittstelle übersetzt werden.

Listing 4 zeigt das Resultat dieser Konvertierung der Transportabschnitte aller Angebote und Nachfragen ist eine alpha-numerische Zeichenkette in folgender Notation:¹³⁷¹

¹³⁷⁰ Vgl. Kapitel 5.2.3, S. 79.

¹³⁷¹ Die in Listing 4 eingefügten Zeilenumbrüche dienen der besseren Lesbarkeit in dieser Forschungsarbeit. In der eigentlichen Implementierung sind sie nicht vorhanden, so dass die Bestandteile der Transportabschnitte in einer einzelnen Zeile notiert werden.


```

"
<UID-SECTION>;
<UID-STATION-START>-<UID-STATION-ZIEL>;
<DATE-FROM>-<DATE-UNTIL>;
<MAX-VOLUME>;
<MAX-MASS>
"

```

Listing 4: Agenten-Notation für Transportabschnitte

Diese Notation ist für Angebots- und Nachfragepräferenzen identisch, da es für die Beschreibung eines Transportabschnitts unerheblich ist, ob dieser Teil eines Angebots oder einer Nachfrage ist.

Tabelle 13 zeigt eine Erläuterung der einzelnen Bestandteile der Notation.

Platzhalter	Erläuterung
<UID-SECTION>	eindeutige Bezeichnung des Transportabschnitts
<UID-STATION-START>	eindeutige Bezeichnung des Start-Haltepunkts
<UID-STATION-ZIEL>	eindeutige Bezeichnung des Ziel-Haltepunkts
<DATE-FROM>	frühestmögliche Abfahrt am Start-Haltepunkt
<DATE-UNTIL>	spätestzulässige Ankunft am Ziel-Haltepunkt
<MAX-VOLUME>	maximal transportierbares/zulässiges Volumen
<MAX-MASS>	maximal transportierbares/zulässiges Gewicht

Tabelle 13: Erläuterung zur Agenten-Notation für Transportabschnitte¹³⁷²

Die so notierten Transportabschnitte sind Teil von Angeboten oder Nachfragen, welche von der Schnittstelle als Gebote formuliert werden müssen.

Listing 5 zeigt die einheitliche Notation, in welcher alle Gebote für Angebote und Nachfragen notiert werden. Zur leichteren Lesbarkeit sind die Transportabschnitte nur verkürzt aufgeführt.

¹³⁷² Quelle: eigene Darstellung.

```

{
  items:  [
    "<UID-SECTION1>;... ",
    "<UID-SECTION2>;... ",
    ...
  ],
  xor:    <ALT-WERT>,
  price:  <PREIS>
}

```

Listing 5: Agenten-Notation für Präferenzen

Tabelle 14 zeigt eine Erläuterung der einzelnen Bestandteile der Notation.

Feld	Platzhalter	Erläuterung
items	-	Liste aller Transportabschnitte
xor	<ALT-WERT>	alpha-numerische Variante des ALT-Werts
price	<PREIS>	Zahlungsbereitschaft in EUR (negativ notiert für Angebote)

Tabelle 14: Erläuterung zur Agenten-Notation für Präferenzen¹³⁷³

7.4.2.3 Beispiel

Das folgende Beispiel illustriert diese Notation für Angebote und Nachfragen.

Ein Teilnehmer bietet

- für den 01. Dezember 2017 zwischen 13:00 und 14:00 Uhr¹³⁷⁴
- flexible Transporte im Umland der Stadt Essen
- im Radius von 10 km um den festen Haltepunkt „Bahnhof Essen“
- für Güter von maximal 100 Kubikmetern Volumen sowie
- maximal 50 Tonnen Gewicht an.

Die Schnittstelle ermittelt nun, dass ein Teilnehmer in diesem Zeitraum einen Transport zwischen einer Postanschrift in Wattenscheid-Höntrop und dem Bahnhof der Stadt Gelsenkirchen (IBNR: 8000118) nachfragt. Sowohl Start- als auch Ziel-Haltepunkt befinden sich im Umkreis von 10 km um Essen und fallen somit geografisch und zeitlich in den Rahmen des Angebots des Anbieters.

¹³⁷³ Quelle: eigene Darstellung.

¹³⁷⁴ Wie bereits beschrieben, werden alle Zeitangaben in koordinierter Weltzeit (UTC) notiert.

Listing 7 zeigt die Notation, die sich in diesem Fall für den Transportabschnitt ergibt:

```
"
0001;
wayp:0000003-rail:8000118;
20171201.1300-20171201.1400;
100.0;
50.0
"
```

Listing 6: Beispielhafter Transportabschnitt in Agenten-Notation (1)

Tabelle 15 zeigt analog zur Tabelle 13 eine Erläuterung der einzelnen Bestandteile.

Wert	Erläuterung
0001	Bezeichnung des Transportabschnitts (Transportabschnitt Nr. 1 innerhalb des IDE-Projekts)
wayp:0000003	Bezeichnung des Start-Haltepunkts (Wegpunkt Nr. 3 innerhalb des IDE-Projekts)
rail:8000118	Bezeichnung des Ziel-Haltepunkts (Schienenbahnhof mit IBNR 8000118)
20171201.1300	frühestmögliche Abfahrt am Start-Haltepunkt (01.12.2017 13:00)
20171201.1400	spätestzulässige Ankunft am Ziel-Haltepunkt (01.12.2017 14:00)
100.0	maximal transportierbares/zulässiges Volumen (100 Kubikmeter)
50.0	maximal transportierbares/zulässiges Gewicht (50 Tonnen)

Tabelle 15: Erläuterung zum beispielhaften Transportabschnitt in Agenten-Notation¹³⁷⁵

Im vorliegenden Beispiel wird ein flexibler Transport im Umkreis um einen festen Haltepunkt betrachtet. Ein Transport zwischen zwei festen Haltepunkten würde sich von diesem Beispiel nur durch die Notation des Start-Haltepunkts unterscheiden.

Würde derselbe Transportabschnitt nicht von einer Postanschrift in Wattenscheid-Höntrop, sondern bspw. vom Bahnhof in Essen (IBNR: 8000098) ausgehen, wäre der Transportabschnitt wie folgt notiert:

¹³⁷⁵ Quelle: eigene Darstellung.

```

"
0002;
rail:8000098-rail:8000118;
20171201.1300-20171201.1400;
100.0;
50.0
"

```

Listing 7: Beispielhafter Transportabschnitt in Agenten-Notation (2)

In der Notation dieses zweiten Beispiels weichen nur zwei Werte ab:

1. die eindeutige Bezeichnung des Transportabschnitts (UID), da es sich um einen anderen Transportabschnitt als im ersten Beispiel handelt, sowie
2. die Bezeichnung des Start-Haltepunkts.

Im bereits beschriebenen Format werden Angebote und Nachfragen als Gebote formuliert. Listing 8 zeigt dies beispielhaft.¹³⁷⁶

```

{
  items: [
    "0002;..."
  ],
  xor: "offer42",
  price: -10000.0
}

```

Listing 8: Beispielhafte Anbieterpräferenz in Agenten-Notation

Zur leichteren Lesbarkeit erläutert Tabelle 16 das oben notierte Angebot analog zu Tabelle 14.

Wert	Erläuterung
["0002;..."]	Liste von Transportabschnitten, welche Teile des Gebots sind
offer42	alpha-numerische Variante des ALT-Werts
-10000.0	Zahlungsbereitschaft in EUR (negativ notiert für Angebote)

Tabelle 16: Erläuterung zur beispielhaften Anbieterpräferenz in Agenten-Notation¹³⁷⁷

¹³⁷⁶ Zur leichteren Lesbarkeit sind die Transportabschnitte wiederum nur mit ihrer UID aufgeführt.

¹³⁷⁷ Quelle: eigene Darstellung.

Listing 9 zeigt analog die Notation eines Angebots mit mehreren Transportabschnitten.

```
{
  items: [
    "0003;...",
    "0004;...",
    "0005;..."
  ],
  xor: "offer42",
  price: -10000.0
}
```

Listing 9: Beispielhafte Anbieterpräferenz mit mehreren Transportabschnitten in Agenten-Notation

Für den Fall, dass mehrere Angebote mit demselben ALT-Wert existieren, kann nur eines von ihnen in der Versteigerung den Zuschlag erhalten.

Listing 10 zeigt ein Beispiel, in welchem entweder das Angebot eines einfachen Transports oder das Bündel aus vier Transportabschnitten den Zuschlag erhalten kann.

```
{
  items: [
    "0001;..."
  ],
  xor: "offer42",
  price: -10000.0
},
{
  items: [
    "0006;...",
    "0007;...",
    "0008;...",
    "0009;..."
  ],
  xor: "offer42",
  price: -10000.0
}
```

Listing 10: Beispielhafte Anbieterpräferenz mit alternativen Angeboten in Agenten-Notation

Die Agenten handeln in den Ad-hoc-Auktionen somit Repräsentationen von Transportdienstleistungen in Form von Kombinationen alpha-numerischer Zeichenketten, denen ein ALT-Wert und ein Preis zugeordnet sind.

7.4.3 Leistungsumfang

7.4.3.1 Kommunikationsmechanismen

Wie bereits in Kapitel 6.2.2 beschrieben, greift die vorliegende Implementierung der Agenten-Software zur Kommunikation auf die Protokollfamilie „TCP/IP“¹³⁷⁸ zurück.

Dies ermöglicht die Adressierung von Ressourcen mit Hilfe einer Adresse, welche aus einem Hostnamen (bspw. „demo.afex-system.org“) oder einer IP-Adresse (bspw. „127.0.0.1“) sowie einer Portnummer (bspw. 1025) besteht.

Eine vollständige Adresse lautet also bspw. „127.0.0.1:1025“. Nur mit Hilfe dieser Informationen können Agenten sich untereinander im Internet kontaktieren.

Die Agenten-Software startet einen TCP-Server, zu dem sich andere Agenten als Clients verbinden können. Jeder Agent fungiert somit zur selben Zeit für einige Verbindungen als Client und für andere als Server.¹³⁷⁹ Die Verbindung ist in der vorliegenden Implementierung weder verschlüsselt noch in anderer Art geschützt. Aus technischer Sicht lässt sich eine Verschlüsselung der übertragenen Daten jedoch aufgrund des modularen Aufbaus der Agenten-Software mit geringem Aufwand realisieren, da lediglich das für die Übertragung der Daten zuständige Modul überarbeitet werden muss.¹³⁸⁰

Dieser Mechanismus wird anschließend genutzt, um mit bekannten anderen Agenten Nachrichten auszutauschen. Eine Übersicht aller den Agenten zur Verfügung stehenden Nachrichten findet sich im Anhang.¹³⁸¹

7.4.3.2 Kontaktweiterleitung

Die in Kapitel 6.2.3 beschriebene Funktion des Funkfeuer-Agenten wird im dezentralen Netzwerk von allen Agenten parallel zu ihren weiteren Tätigkeiten wahrgenommen. Alle Agenten leiten die von ihnen gefundenen Kontakte an alle mit ihnen in Kontakt stehenden Agenten weiter.¹³⁸²

Kontaktiert ein Agent den TCP-Server eines anderen Agenten, so sendet dieser ihm eine Nachricht mit allen ihm bekannten Adressen von anderen Agenten. Der initial anfragende

¹³⁷⁸ Vgl. WRIGHT/STEVENS (1995).

¹³⁷⁹ Vgl. Anhang, S. 367.

¹³⁸⁰ Vgl. Kapitel 10.2.3.3, S. 249.

¹³⁸¹ Vgl. Anhang, S. 341.

¹³⁸² Vgl. Anhang, S. 354.

Agent antwortet auf diese Nachricht mit einer eigenen Nachricht, die analog alle ihm bekannten Adressen anderer Agenten enthält.

Diese transparente, jedoch ungeschützte Weitergabe von Kontaktdaten ließe sich im Rahmen einer Weiterentwicklung dahingehend einschränken, dass nicht zu allen Kontakt anfragenden Agenten ein aktiver Kontakt aufgebaut wird.¹³⁸³

7.4.3.3 Präferenzoffenlegung und -weiterleitung

Die Agenten kennen ihre eigenen Handelspräferenzen durch den in Kapitel 7.4.2.1 beschriebenen Prozess.¹³⁸⁴ Parallel zur reinen Kontaktweitergabe kommunizieren die Agenten ihre Handelspräferenzen an alle Agenten, die sie neu kennenlernen.

Hierdurch verfügt jeder Agent über Kenntnis der Handelspräferenzen seiner Kontakte. Diese Handelspräferenzen ermöglichen anschließend die gezielte Ansprache einzelner Agenten zur Bildung einer Gruppe (im Falle kompatibler Präferenzen).

Wie bei der Kontaktweitergabe im Kapitel zuvor besteht auch bei der Präferenzoffenlegung ein wichtiges Weiterentwicklungspotential darin, die Offenlegung der Präferenzen im Vorfeld auf bestimmte anfragende Agenten zu beschränken.¹³⁸⁵

¹³⁸³ Vgl. Kapitel 10.2.3.3, S. 249.

¹³⁸⁴ Vgl. Kapitel 7.4.2.1, S. 216.

¹³⁸⁵ Vgl. Kapitel 10.2.2.4, S. 245.

7.4.3.4 Implementiertes Auktionsmodell

Das implementierte Auktionsmodell ist eine triviale Erweiterung des in Kapitel 5.3.3.6 vorgestellten, allgemeinen Modells, im Folgenden als „Grundmodell“ bezeichnet:¹³⁸⁶

$$\begin{aligned}
 \max \quad & \sum_{j=1}^n p_j x_j \\
 \text{s.t.} \quad & \sum_{j=1}^n \lambda_j^k x_j \leq 0 \quad k = 1, 2, \dots, m \\
 & \sum_{j=1}^n \alpha_j^a x_j \leq 1 \quad a = 1, 2, \dots, n \\
 & \sum_{j=1}^n x_j \geq 2 \\
 & \sum_{j=1}^n p_j x_j \geq 0 \\
 & p_j \in \mathbb{R} \quad j = 1, 2, \dots, n \\
 & x_j \in \{0, 1\} \quad j = 1, 2, \dots, n \\
 & \lambda_j^k \in \{-1, 0, 1\} \quad j = 1, 2, \dots, n \\
 & \quad \quad \quad k = 1, 2, \dots, m \\
 & \alpha_j^a \in \{0, 1\} \quad j = 1, 2, \dots, n \\
 & \quad \quad \quad a = 1, 2, \dots, n
 \end{aligned}$$

Wie bereits beim Grundmodell wird der Auktionsmehrwert maximiert. Die ermittelte Allokation muss sich aus der Kombination von mindestens 2 Geboten ergeben und einen nicht negativen Auktionsmehrwert erzielen, d.h. die gesamte maximale Zahlungsbereitschaft muss mindestens der Höhe der gesamten minimalen Zahlungsforderung entsprechen.

Durch die Gruppenbildung wurde determiniert, welche Güter k aus der Menge $M = \{1, 2, \dots, m\}$ in den insgesamt n Geboten der Ad-hoc-Auktion berücksichtigt werden. Ein Gebot ist innerhalb des Auktionsmodells definiert als $B_j = \{(\lambda_j^1, \lambda_j^2, \dots, \lambda_j^m), (\alpha_j^1, \alpha_j^2, \dots, \alpha_j^n), p_j\}$, wobei λ_j^k das Gebot entweder als Angebot von (-1) oder Nachfrage nach Gut k (1) kennzeichnet oder die Indifferenz des Teilnehmer gegenüber der Veräußerung/dem Erwerb von Gut k (0) zum Ausdruck bringt. p_j ist der Preis des Güterbündels. Er ist analog zur Binärvariable λ_j^k negativ notiert, wenn es sich bei dem Gebot um ein Angebot für das spezifizierte Güterbündel handelt.

Das Modell beinhaltet zusätzlich die in Kapitel 6.4.2 eingeführten ALT-Werte.¹³⁸⁷

¹³⁸⁶ Vgl. Kapitel 5.3.3.6, S. 109.

¹³⁸⁷ Vgl. Kapitel 6.4.2, S. 134.

α_j^a ist eine Binärvariable, die widerspiegelt, welche Gebote sich auf Basis ihrer ALT-Werte gegenseitig ausschließen. So kann definiert werden, dass sich die Gebote B_a und B_j gegenseitig ausschließen, d.h. sie können nicht gleichzeitig in der Allokation enthalten sein ($\alpha_j^a = 1$). Alternativ kann definiert werden, dass beide Gebote in der Allokation enthalten sein können ($\alpha_j^a = 0$).

Das folgende Beispiel soll das Modell verdeutlichen:

- **(Gebot 1)** Anbieter 1 verkauft Gut A für 30 EUR.
- **(Gebot 2)** Anbieter 1 verkauft die Güter B und C in Kombination für 50 EUR
- **(Gebot 3)** *oder* Anbieter 1 verkauft Gut B für 40 EUR
- **(Gebot 4)** *oder* Anbieter 1 verkauft Gut C für 30 EUR.
- **(Gebot 5)** Nachfrager 1 würde Gut A für 30 EUR kaufen.
- **(Gebot 6)** Nachfrager 2 würde Güter $B + C$ für 60 EUR kaufen.
- **(Gebot 7)** Nachfrager 3 würde Güter $A + B$ für 80 EUR kaufen.

Tabelle 17 zeigt die resultierenden Werte.

	λ_j^A	λ_j^B	λ_j^C	p_j	ALT_j	x_j
Gebot 1	-1	0	0	-30	1	1
Gebot 2	0	-1	-1	-50	2	1
Gebot 3	0	-1	0	-40	2	0
Gebot 4	0	0	-1	-30	2	0
Gebot 5	1	0	0	30	3	1
Gebot 6	0	1	1	60	4	1
Gebot 7	1	1	0	80	5	0

Tabelle 17: Zahlenbeispiel für eine Börse mit ALT-Werten¹³⁸⁸

Mit Hilfe der ALT-Werte kann eine Matrix aus Binärwerten erstellt werden, die angibt, welche Gebote sich jeweils gegenseitig ausschließen. Tabelle 18 zeigt diese Matrix für das obige Beispiel.

¹³⁸⁸ Quelle: eigene Darstellung.

	Gebot 1	Gebot 2	Gebot 3	Gebot 4	Gebot 5	Gebot 6	Gebot 7
Gebot 1	0	0	0	0	0	0	0
Gebot 2	0	0	1	1	0	0	0
Gebot 3	0	1	0	1	0	0	0
Gebot 4	0	1	1	0	0	0	0
Gebot 5	0	0	0	0	0	0	0
Gebot 6	0	0	0	0	0	0	0
Gebot 7	0	0	0	0	0	0	0

Tabelle 18: Zahlenbeispiel für eine ALT-Wert-Matrix¹³⁸⁹

Im Gegensatz zum Grundmodell gibt jeder Teilnehmer beliebig viele Gebote ab, so dass die Anzahl der Teilnehmer nicht der Anzahl der Gebote entsprechen muss.

In diesem Beispiel erhalten die Gebote 1, 2, 5 und 6 den Zuschlag. Anbieter 1 würde die Güter A, B und C an die Nachfrager 1 und 2 verkaufen, da der Auktionsmehrwert für diese Allokation maximal ist.

7.4.3.5 Durchführung von Ad-hoc-Auktionen

Die Lösung des im vorangegangenen Kapitel vorgestellten kombinatorischen Problems fällt in den Bereich der linearen Programmierung (teils auch „lineare Optimierung“). Die lineare Programmierung befasst sich mit der Minimierung oder Maximierung einer linearen Zielfunktion über einer Menge zulässiger Lösungen, die durch lineare Gleichungen und Ungleichungen eingeschränkt ist. Sie eignet sich zur Lösung des definierten Auktionsmodells. Weitergehende Einführungen in die lineare Programmierung finden sich z.B. bei OKI sowie BAZARAA et al.¹³⁹⁰

Zur Lösung dieser ganzzahligen linearen Optimierungsprobleme kann nach der Problembeschreibung in Modellform eine spezielle Software, sog. „Solver“, eingesetzt werden, um die optimale Lösung für eine gegebene Menge an Variablen zu ermitteln.¹³⁹¹ Beispiele für diese Art von Software sind das kommerzielle CPLEX des amerikanischen Unternehmens IBM sowie das frei verfügbare, kostenlose Open-Source-Projekt GNU Linear Programming Kit (GLPK).¹³⁹²

¹³⁸⁹ Quelle: eigene Darstellung.

¹³⁹⁰ Vgl. OKI (2012), S. 5 ff., sowie BAZARAA et al. (2010), S. 1 ff.

¹³⁹¹ Vgl. KASPERSKI/KULEJ (2009), S. 2.

¹³⁹² Vgl. GEARHART et al. (2013), S. 12 ff., sowie MEINDL/TEMPL (2012), S. 2 f.

In einem Vergleich zwischen kommerziellen und kostenlosen Solvern von MEINDL und TEMPL wird GLPK als bester kostenloser Solver für kleine Probleme ermittelt.¹³⁹³ Obwohl GLPK auch für das Lösen kleiner Probleme langsamer ist als die in der Studie betrachteten kommerziellen Alternativen, ist der Zeitunterschied zum Erreichen einer optimalen Lösung für diese Problemgrößen vernachlässigbar.¹³⁹⁴ Diverse Veröffentlichungen nutzen aus diesen Gründen GLPK zum Lösen unterschiedlicher Optimierungsprobleme.¹³⁹⁵

In der prototypischen Implementierung wird zur Lösung des kombinatorischen Problems auf GLPK zurückgegriffen. GLPK verwendet das Branch-and-Bound-Verfahren.¹³⁹⁶

Die Trivialität des Modells und die Anwendung eines einfachen Solvers sind der prototypischen Natur der Implementierung zuzuschreiben. Eine Weiterentwicklung der Agenten-Software könnte eine Vielzahl an möglichen Algorithmen und Modellen nutzen. Der modulare Aufbau des Prototyps eignet sich zur schrittweisen Verbesserung dieser Aspekte. Verschiedene Beispiele hierzu werden in Kapitel 10.2 aufgezeigt.¹³⁹⁷

7.4.3.6 Verarbeitung der Auktionsergebnisse im Netzwerk

Nach Abschluss einer Auktion werden allen beteiligten Agenten die Auktionsergebnisse per Nachricht mitgeteilt. Hat eines oder hat mehrere ihrer Gebote den Zuschlag erhalten, so werden die diesen Geboten zugehörigen Präferenzen annulliert. Durch diese Vorgehensweise ist sichergestellt, dass die Agenten zum einen die Nachricht erhalten haben, dass ihre Präferenzen erfüllt wurden, und sie zum anderen nicht mehr in Verhandlungen (andere Gruppen) über diese Präferenzen eintreten.

Möglichkeiten zur digitalen Signierung und Persistierung der Ergebnisse stellen ein wichtiges Weiterentwicklungspotential dar und werden in Kapitel 10.2.3.7 aufgezeigt.¹³⁹⁸

¹³⁹³ Vgl. MEINDL/TEMPL (2012), S. 11.

¹³⁹⁴ Vgl. MEINDL/TEMPL (2012), S. 11.

¹³⁹⁵ Vgl. LEAL-TAIXÉ et al. (2011), CHANG et al. (2009), KASPERSKI/KULEJ (2009), sowie TAGHAVI/SIEGEL (2007).

¹³⁹⁶ Vgl. CRAINIC et al. (2006), S. 13.

¹³⁹⁷ Vgl. Kapitel 10.2, S. 244.

¹³⁹⁸ Vgl. Kapitel 10.2.3.7, S. 250.

8 Installation und Betrieb des Softwareprototyps

8.1 Vorbemerkungen

Zur besseren Einordnung der Praxistauglichkeit des implementierten Softwareprototyps wird im Folgenden ein kurzer Überblick gegeben, welche technischen und personellen Anforderungen an die Installation und den Betrieb des Softwareprototyps gestellt werden.

Es sei an dieser Stelle nochmals explizit darauf hingewiesen, dass der vorliegende Softwareprototyp mit dem Ziel entwickelt wurde, Gegenstand und Werkzeug der weiteren Forschung und betriebswirtschaftlichen Anforderungsanalyse im Bereich „Online-Frachtenbörsen für den Güterverkehr“ zu sein. Aus diesem Grunde wurde eine internetbasierte Benutzeroberfläche geschaffen, so dass potentielle Nutzer den Softwareprototyp erleben können, ohne die Software auf ihren Computer herunterladen und installieren zu müssen. Hierdurch ist zur Erprobung des Prototyps lediglich der Zugriff auf einen Computer mit Internetanschluss und Internetbrowser notwendig.

Doch die Entscheidung, den Prototyp als internetbasiertes System zu implementieren, bedingt auf der anderen Seite, dass zur Nutzung dieses Systems ein Internetserver betrieben werden muss. Es ist demnach zwar möglich, den Prototyp ohne IT-Kenntnisse zu nutzen, zur erstmaligen Einrichtung einer Testinstanz wird jedoch eine Person mit weitergehenden IT-Kenntnissen benötigt.¹³⁹⁹

Diese Abwägung wurde vor dem Hintergrund getroffen, dass der Softwareprototyp, wenn er zur weiteren Erprobung und Demonstration der zuvor vorgestellten Konzepte eingesetzt wird, nur einmalig eingerichtet werden muss, aber anschließend von beliebig vielen Personen ohne weitergehende IT-Kenntnisse genutzt werden kann, da die Software nicht auf weiteren Computern installiert werden muss.

¹³⁹⁹ Dies stellt eine Einstiegshürde dar. Die leichtere Installation dieser Internetserver stellt ein identifiziertes Weiterentwicklungspotential der Software dar. Der Grund, warum die leichtere Installation der Agenten-Software keine Priorität im Rahmen dieser Forschungsarbeit besitzt, liegt darin, dass die gewählten Implementierungstechniken eine leichtere Installation – bspw. durch Bereitstellung einer ausführbaren EXE-Datei – nicht begünstigen. Die Implementierungstechniken verstärkt danach auszuwählen, ob sie einfachere Installationsroutinen ermöglichen, schien wiederum den Zielen dieser Forschungsarbeit nicht dienlich. Vgl. Kapitel 10.2.3.1, S. 248.

8.2 Installation des Softwareprototyps

Der vorliegende Softwareprototyp ist eine vorwiegend in der Programmiersprache Ruby implementierte Software. Um eine Anbindung an den verwendeten Solver GLPK zu ermöglichen, wird für das Auktionsmodul zusätzlich die Programmiersprache Python verwendet. Es wird empfohlen, den Softwareprototyp auf debian-basierten Linux-Betriebssystemen zu installieren.¹⁴⁰⁰ Ruby und Python müssen jeweils in der Version 2 auf dem Computer vorinstalliert sein, um den Prototyp betreiben zu können. In Ruby muss das Programm „bundler“ installiert sein, um den Paketmanager RubyGems nutzen zu können.¹⁴⁰¹

Zur Nutzung der integrierten Entwicklungsumgebung ist zudem die Installation der Datenbank SQLite zur Speicherung der IDE-Daten sowie des Schlüssel-Werte-Speichers Redis zur Verwaltung der durch die IDE generierten Startparameter für die Agenten erforderlich. Die Benutzeroberfläche der integrierten Entwicklungsumgebung wurde mit Hilfe des Webframeworks „Ruby on Rails“¹⁴⁰² implementiert.

Der Solver GLPK¹⁴⁰³ muss zusammen mit der Python-Bibliothek PymProg¹⁴⁰⁴ installiert sein, um die Agenten in die Lage zu versetzen, Auktionen durchzuführen.

Eine detaillierte, in Fachsprache verfasste Anleitung zur Installation sowie zum Betrieb des Softwareprototyps in englischer Sprache findet sich im Anhang.¹⁴⁰⁵

8.3 Betrieb des Softwareprototyps

Zur Implementierung der Benutzeroberfläche des Softwareprototyps wurde auf freie und quelloffene Dienste zurückgegriffen, die in Bezug auf ihre visuelle Qualität und Ausfallsicherheit nicht mit kommerziellen Angeboten zu vergleichen sind. Dies betrifft insbesondere die Kartendarstellung innerhalb der IDE. Es wurde mit Hilfe von kostenlos und frei verfügbaren Diensten eine Benutzeroberfläche gestaltet, die in ihrem Erscheinungsbild an professionelle Angebote wie bspw. „Google Maps“¹⁴⁰⁶ erinnert. Da die kostenfreien Kartendienste nicht über ähnlich redundante und ausfallsichere Systeme verfügen wie

¹⁴⁰⁰ Der Softwareprototyp wurde unter dem Betriebssystem Ubuntu entwickelt. Weitere Informationen sind im Internet verfügbar. Vgl. UBUNTU (2017).

¹⁴⁰¹ Weitere Informationen sind im Internet verfügbar. Vgl. RUBYGEMS (2017).

¹⁴⁰² Weitere Informationen sind im Internet verfügbar. Vgl. RUBYONRAILS (2017).

¹⁴⁰³ Weitere Informationen sind im Internet verfügbar. Vgl. GNU (2017).

¹⁴⁰⁴ Weitere Informationen sind im Internet verfügbar. Vgl. PYMPROG (2017).

¹⁴⁰⁵ Vgl. Anhang, S. 283.

¹⁴⁰⁶ Google Maps ist ein von dem Unternehmen Google betriebener Kartendienst, der neben kontextuellen Informationen zu öffentlichen Plätzen, Gebäuden und Betrieben zahlreicher Branchen auch einen Routenplaner enthält. Weitere Informationen sind im Internet verfügbar. Vgl. GOOGLE (2017).

kommerzielle Anbieter, kann es vereinzelt zu Dienstaussfällen kommen, so dass die grafische Karte, auf der die Informationen des Softwareprototyps dargestellt werden, nicht lädt.

Wenngleich dieser Umstand den finanziellen und personellen Einschränkungen einer Forschungsarbeit geschuldet ist, so soll auch festgestellt werden, dass die in dieser Forschungsarbeit beschriebenen Funktionen des Softwareprototyps von diesen Unsicherheiten nicht betroffen sind. Sie wirken sich lediglich auf die Benutzerführung und die Benutzerfreundlichkeit aus.

Während der Implementierung musste der Nutzen des Eindrucks einer vertraut und professionell wirkenden Software gegen das Risiko eines Ausfalls des Kartendienstes abgewägt werden.¹⁴⁰⁷ Bei dieser Abwägung überwog der erwartete Zusatznutzen, den eine ansprechend und vertraut wirkende Benutzeroberfläche auf potentielle Endnutzer des Softwareprototyps haben kann.

Zum Betrieb des Softwareprototyps ist bewusst keine anspruchsvolle Hardwarekonfiguration erforderlich. Der Ressourcenbedarf der Software steigt jedoch mit der Komplexität der simulierten Szenarien, so dass auf Serversystemen mit weniger Arbeitsspeicher weniger komplexe Szenarien simuliert werden können als auf solchen mit mehr Arbeitsspeicher.¹⁴⁰⁸

Beim Betrieb des Softwareprototyps über das Internet ist jedoch zu beachten, dass die Software aufgrund der prototypischen Natur der Implementierung keinerlei Sicherheitsvorkehrungen in Form von Zugriffsbeschränkungen bietet. Wird der Softwareprototyp unter einer Adresse abrufbar gemacht, so ist er jeder Person mit Kenntnis dieser Adresse zugänglich. Es empfiehlt sich daher, im Falle des öffentlichen Betriebs die Benutzeroberfläche der integrierten Entwicklungsumgebung mit einer sog. „HTTP basic authentication“ zu schützen, um den Zugriff durch Dritte zu limitieren.

¹⁴⁰⁷ Es sei an dieser Stelle anekdotisch angemerkt, dass ein solcher Ausfall während der gesamten Implementierungsphase dieser Forschungsarbeit nur einmal auftrat.

¹⁴⁰⁸ Während der Implementierung wurde ein Server mit 8 GB Arbeitsspeicher verwendet. Die in der vorliegenden Arbeit beschriebenen Simulationen konnten hiermit ohne Einschränkungen durchgeführt werden.

9 Simulation des Multi-Agenten-Systems

9.1 Aufbau der Szenarien

Zur Zusammenführung der aufgestellten Konzepte für eine agentenorientierte Online-Frachtenbörse und der beschriebenen prototypischen Implementierung werden in diesem Kapitel Szenarien beschrieben werden, welche als Anwendungsbeispiele mit der IDE reproduziert werden können.

Alle beschriebenen Szenarien werden auf der Kartenvorlage „Nordrhein-Westfalen (vereinfacht)“ angelegt. Es handelt sich hierbei um einen vereinfachten Kartenausschnitt der Kartenvorlage „CODE24“.

In jedem dieser Beispiele wird deutlich, dass die nachfragenden Agenten nur an der vollständigen kombinatorischen Erfüllung aller ihrer Teilgebote interessiert sind. Eine klassische Auktionsform würde in diesen Szenarien für die Unsicherheit sorgen, dass der bietende Agent nur einen Teil der präferierten Transportdienstleistungen ersteigert.

Für jedes Szenario werden die anzulegenden Agenten sowie Angebote für und Nachfragen nach Transportdienstleistungen beschrieben.

Zur leichteren Lesbarkeit wird dabei folgende Konvention verwendet:

Alle Angebote und Nachfragen ...

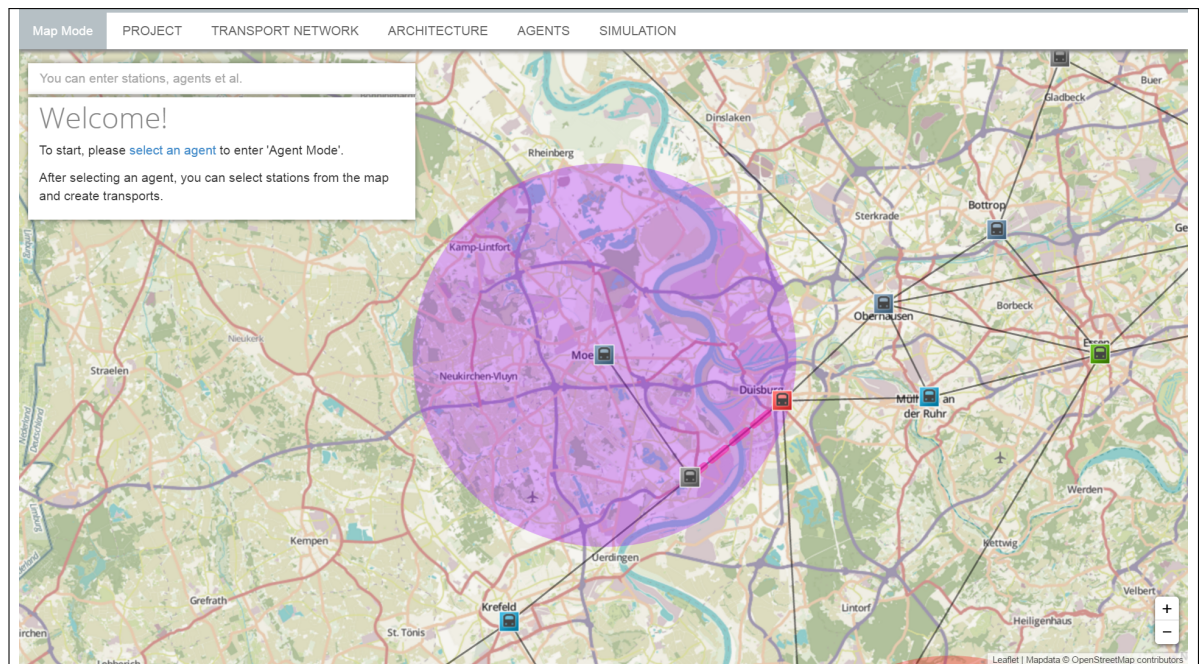
- finden am selben Tag statt (bspw. am 1. Januar 2017, lediglich die Uhrzeit variiert),
- haben dieselben Angaben zum Gewicht (bspw. 50),
- zum Volumen (bspw. 100) und
- zur Güterkategorie (bspw. 01000000).

9.2 Szenario: Spezialfall 1

Als erstes Szenario wird eine Verbindung zwischen zwei festen Haltepunkten betrachtet, welche durch das Angebot einer Verbindung zwischen zwei flexiblen Haltepunkten bedient werden kann. Dieses Szenario greift Spezialfall 1 aus Kapitel 6.3.3 auf und erweitert es um den Aspekt miteinander konkurrierender Agenten.¹⁴⁰⁹

Abbildung 48 zeigt das Szenario im Karten-Modus.

¹⁴⁰⁹ Vgl. Kapitel 6.3.3, S. 127.

Abbildung 48: Kartenansicht im Szenario „Spezialfall 1“¹⁴¹⁰

Die für das Szenario anzulegenden Agenten sind:

- Anbieter 1 bietet Lkw-Transporte im Umkreis von 10 km um Moers an.
- Anbieter 2 bietet einen Zug-Transport von Reinhausen nach Duisburg Hbf an.
- Nachfrager 1 fragt einen Transport von Reinhausen nach Duisburg Hbf nach.
- Nachfrager 2 fragt einen Transport von Moers nach Duisburg Hbf nach.

In Abhängigkeit von den Preisen/Zahlungsbereitschaften der Anbieter und Nachfrager substituieren die Lkw-Transporte von Anbieter 1 die Zugverbindung zwischen Reinhausen und Duisburg Hbf von Anbieter 2. Die Nachfrage von Moers nach Duisburg Hbf kann nur durch den Lkw-Transport von Anbieter 1 gedeckt werden.

Tabelle 19 am Ende des Kapitels beschreibt die anzulegenden Angebote für und Nachfragen nach Transportdienstleistungen.

In dieser Konfiguration erhalten die Gebote 2 und 3 (von Anbieter 2 und Nachfrager 1) den Zuschlag.

Variationen des Szenarios

- Wird die Preisvorstellung von Anbieter 1 für Gebot 1 von 1.100 auf 950 EUR gesenkt, so erhalten die Gebote 1 und 3 (von Anbieter 1 und Nachfrager 1) den Zuschlag.

¹⁴¹⁰ Quelle: eigene Darstellung.

- Wird stattdessen die Zahlungsbereitschaft von Nachfrager 2 für Gebot 4 von 900 auf 1.100 EUR erhöht, so erhalten die Gebote 1 und 4 sowie 2 und 3 (von Anbieter 1 und Nachfrager 2 sowie Anbieter 2 und Nachfrager 1) den Zuschlag.

9.3 Szenario: Spezialfall 2

Nachfragen können als Start- oder Zielpunkt einen Wegpunkt in Form einer Postanschrift definiert haben.¹⁴¹¹ Das vorliegende Szenario behandelt den Fall, dass eine in dieser Weise definierte Postanschrift in der Schnittmenge sich überschneidender Radien von angebotenen Lkw-Transporten liegt. Dieses Szenario greift Spezialfall 2 aus Kapitel 6.3.3 auf und erweitert es um den Aspekt miteinander konkurrierender Agenten.¹⁴¹²

Abbildung 49 zeigt das Szenario im Karten-Modus.

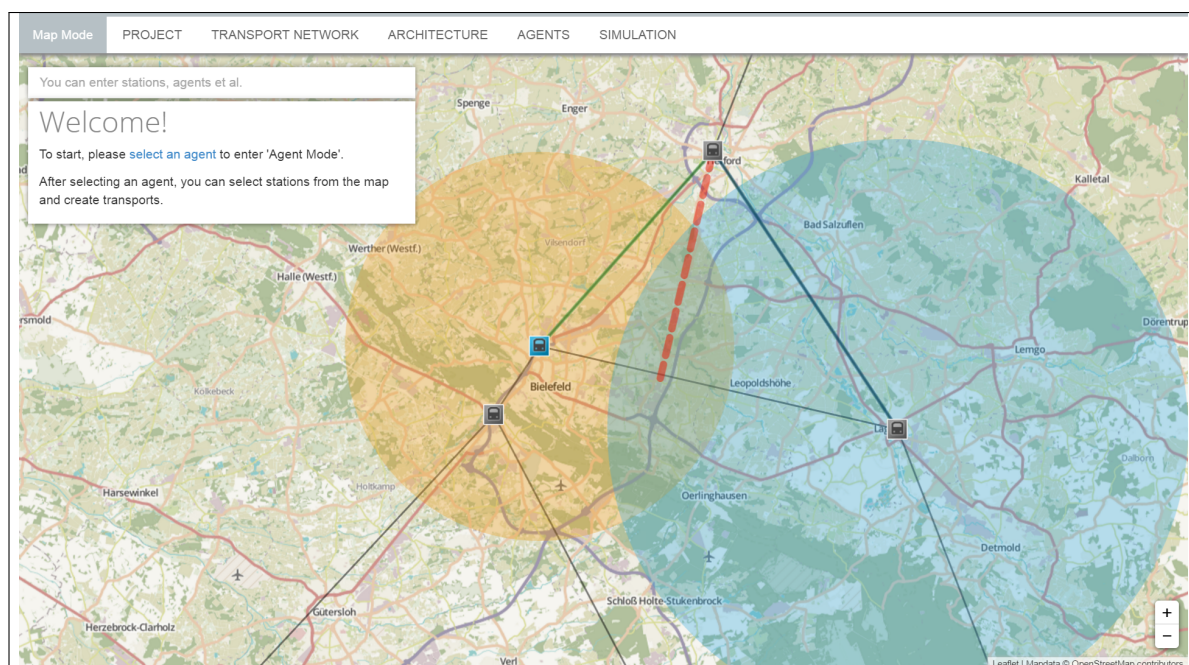


Abbildung 49: Kartenansicht im Szenario „Spezialfall 2“¹⁴¹³

Die für das Szenario anzulegenden Agenten sind:

- Anbieter 1 bietet Lkw-Transporte im Umkreis von 10 km um Moers an.
- Anbieter 2 bietet Lkw-Transporte im Umkreis von 10 km um Moers an.
- Anbieter 3 bietet einen Zug-Transport von Reinhausen nach Duisburg Hbf an.
- Anbieter 4 bietet einen Zug-Transport von Reinhausen nach Duisburg Hbf an.

¹⁴¹¹ Vgl. Kapitel 7.3.3.5, S. 208.

¹⁴¹² Vgl. Kapitel 6.3.3, S. 128.

¹⁴¹³ Quelle: eigene Darstellung.

- Nachfrager 1 fragt einen Transport von Herford zur Postanschrift „Lange Wand, 33719 Bielefeld“ nach.

Tabelle 20 am Ende des Kapitels beschreibt die anzulegenden Angebote für und Nachfragen nach Transportdienstleistungen.

In dieser Konfiguration erhalten die Gebote 1, 3 und 5 (von Anbieter 1 und 3 sowie Nachfrager 1) den Zuschlag.

Variation des Szenarios

- Wird die Preisvorstellung von Anbieter 2 für Gebot 2 von 1.100 auf 900 EUR gesenkt, so erhalten die Gebote 2, 4 und 5 (von Anbieter 2 und 4 sowie Nachfrager 1) den Zuschlag.

9.4 Szenario: Spezialfall 3

Eine Abwandlung des Spezialfalls 3 ergibt einen Fall, in welchem eine aus mehreren Relationen bestehende, feste Strecke durch flexible Haltepunkte substituiert wird. Dieses Szenario greift Spezialfall 3 aus Kapitel 6.3.3 auf und erweitert es um den Aspekt miteinander konkurrierender Agenten.¹⁴¹⁴

Abbildung 50 zeigt das Szenario im Karten-Modus.

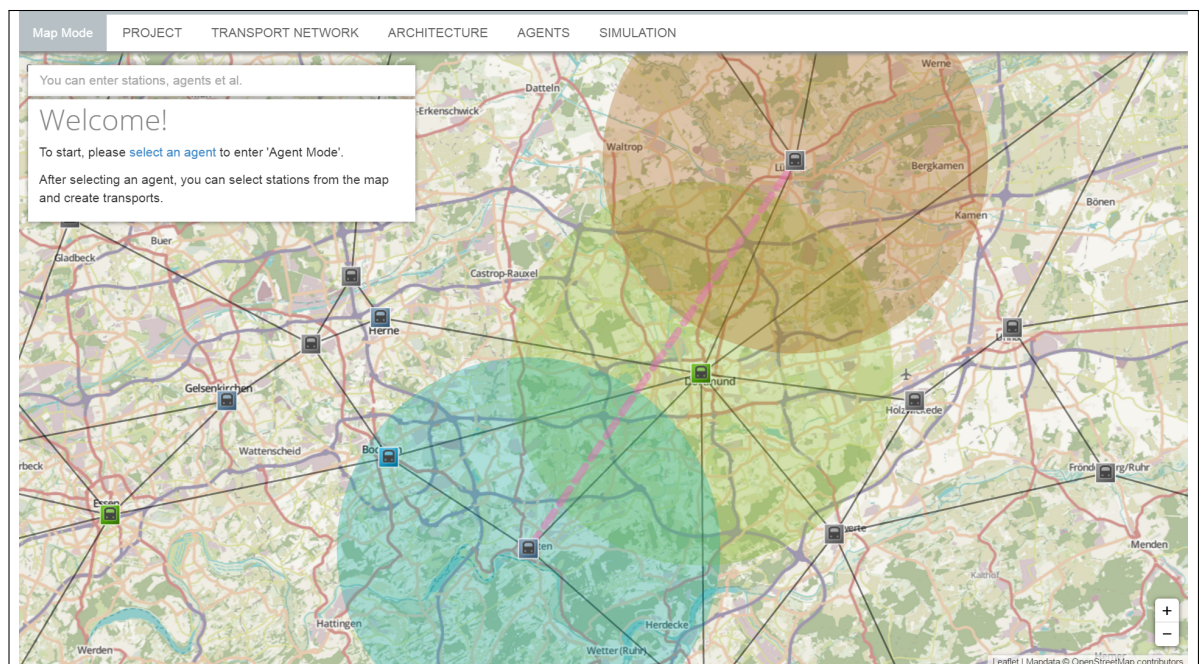


Abbildung 50: Kartenansicht im Szenario „Spezialfall 3“¹⁴¹⁵

¹⁴¹⁴ Vgl. Kapitel 6.3.3, S. 129.

¹⁴¹⁵ Quelle: eigene Darstellung.

Die für das Szenario anzulegenden Agenten sind:

- Anbieter 1 bietet Lkw-Transporte im Umkreis von 10 km um Witten Hbf an.
- Anbieter 2 bietet Lkw-Transporte im Umkreis von 10 km um Dortmund Hbf an.
- Anbieter 3 bietet Lkw-Transporte im Umkreis von 10 km um Lünen Hbf an.
- Anbieter 4 bietet Zug-Transporte von Witten Hbf nach Dortmund Hbf sowie von Dortmund Hbf nach Lünen Hbf an.
- Nachfrager 1 fragt einen Transport von Witten Hbf nach Lünen Hbf nach.

In Abhängigkeit von den Preisen/Zahlungsbereitschaften der Anbieter und Nachfrager substituieren die Lkw-Transporte von Anbieter 1, 2 und 3 die Zugverbindungen zwischen Reinhausen und Duisburg Hbf von Anbieter 2.

Tabelle 21 am Ende des Kapitels beschreibt die anzulegenden Angebote für und Nachfragen nach Transportdienstleistungen.

In dieser Konfiguration erhalten die Gebote 1, 2, 3 und 6 (von Anbieter 1, 2 und 3 sowie Nachfrager 1) den Zuschlag.

Variation des Szenarios

- Wird die Preisvorstellung von Anbieter 4 für die Gebote 4 und 5 von 2.000 auf jeweils 1.000 EUR gesenkt, so erhalten die Gebote 4, 5 und 6 (von Anbieter 4 und Nachfrager 1) den Zuschlag.

9.5 Szenario: Spezialfall 4

Die Zusammenführung der zuvor beschriebenen Szenarien ergibt einen Fall, in welchem eine aus mehreren Relationen bestehende, unvollständige feste Strecke durch einen flexiblen Transport ergänzt wird. Dieses Szenario greift Spezialfall 4 aus Kapitel 6.3.3 auf und erweitert es um den Aspekt miteinander konkurrierender Agenten.¹⁴¹⁶

Abbildung 51 zeigt das Szenario im Karten-Modus.

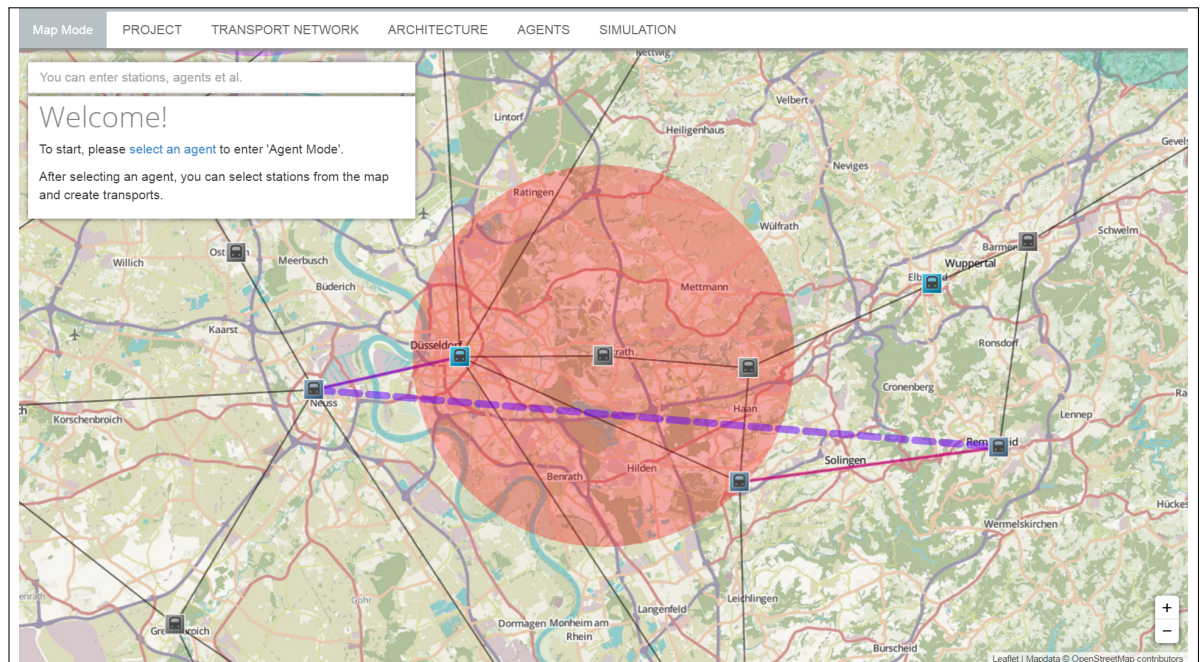


Abbildung 51: Kartenansicht im Szenario „Spezialfall 4“¹⁴¹⁷

Die für das Szenario anzulegenden Agenten sind:

- Anbieter 1 bietet Zug-Transporte von Neuss Hbf nach Düsseldorf Hbf an.
- Anbieter 2 bietet Lkw-Transporte im Umkreis von 10 km um Düsseldorf Hbf an.
- Anbieter 3 bietet Zug-Transporte von Solingen Hbf nach Remscheid Hbf an.
- Nachfrager 1 fragt einen Transport von Neuss Hbf nach Remscheid Hbf nach.

Tabelle 22 am Ende des Kapitels beschreibt die anzulegenden Angebote für und Nachfragen nach Transportdienstleistungen.

In dieser Konfiguration erhalten die Gebote 1, 2, 3 und 4 (von Anbieter 1, 2 und 3 sowie Nachfrager 1) den Zuschlag.

¹⁴¹⁶ Vgl. Kapitel 6.3.3, S. 130.

¹⁴¹⁷ Quelle: eigene Darstellung.

#	Agent	Angebot/Nachfrage	fest/flexibel/-	Startpunkt	Zielpunkt/Radius	frühester Abfahrtszeitpunkt	spätester Ankunftszeitpunkt	Preis/Zahlungsbereitschaft
1	Anbieter 1	Angebot	flexibel	Moers	10 km	10:00:00 UTC	11:00:00 UTC	1.100
2	Anbieter 2	Angebot	fest	Reinhausen	Duisburg Hbf	10:00:00 UTC	11:00:00 UTC	1.000
3	Nachfrager 1	Nachfrage	-	Reinhausen	Duisburg Hbf	10:00:00 UTC	11:00:00 UTC	1.000
4	Nachfrager 2	Nachfrage	-	Moers	Duisburg Hbf	10:00:00 UTC	11:00:00 UTC	900

Tabelle 19: Angebote und Nachfragen im Szenario „Spezialfall 1“

#	Agent	Angebot/Nachfrage	fest/flexibel/-	Startpunkt	Zielpunkt/Radius	frühester Abfahrtszeitpunkt	spätester Ankunftszeitpunkt	Preis/Zahlungsbereitschaft
1	Anbieter 1	Angebot	flexibel	Bielefeld Hbf	10 km	10:00:00 UTC	11:00:00 UTC	1.000
2	Anbieter 2	Angebot	flexibel	Lage (Lippe)	15 km	10:00:00 UTC	11:00:00 UTC	1.100
3	Anbieter 3	Angebot	fest	Herford	Bielefeld Hbf	11:00:00 UTC	12:00:00 UTC	1.000
4	Anbieter 4	Angebot	fest	Herford	Lage (Lippe)	11:00:00 UTC	12:00:00 UTC	1.000
5	Nachfrager 1	Nachfrage	-	Herford	Lange Wand, 33719 Bielefeld	10:00:00 UTC	12:00:00 UTC	2.000

Tabelle 20: Angebote und Nachfragen im Szenario „Spezialfall 2“

#	Agent	Angebot/Nachfrage	fest/flexibel/-	Startpunkt	Zielpunkt/Radius	frühester Abfahrtszeitpunkt	spätester Ankunftszeitpunkt	Preis/Zahlungsbereitschaft
1	Anbieter 1	Angebot	flexibel	Witten Hbf	10 km	10:00:00 UTC	11:00:00 UTC	1.000
2	Anbieter 2	Angebot	flexibel	Dortmund Hbf	10 km	11:00:00 UTC	12:00:00 UTC	1.000
3	Anbieter 3	Angebot	flexibel	Lünen Hbf	10 km	12:00:00 UTC	13:00:00 UTC	1.000
4	Anbieter 4	Angebot	fest	Witten Hbf	Dortmund Hbf	10:00:00 UTC	11:00:00 UTC	2.000
5	Anbieter 4	Angebot	fest	Dortmund Hbf	Lünen Hbf	11:00:00 UTC	12:00:00 UTC	2.000
6	Nachfrager 1	Nachfrage	-	Witten Hbf	Lünen Hbf	10:00:00 UTC	13:00:00 UTC	3.000

Tabelle 21: Angebote und Nachfragen im Szenario „Spezialfall 3“

#	Agent	Angebot/Nachfrage	fest/flexibel/-	Startpunkt	Zielpunkt/Radius	frühester Abfahrtszeitpunkt	spätester Ankunftszeitpunkt	Preis/Zahlungsbereitschaft
1	Anbieter 1	Angebot	fest	Neuss Hbf	Düsseldorf Hbf	10:00:00 UTC	11:00:00 UTC	1.000
2	Anbieter 2	Angebot	flexibel	Düsseldorf Hbf	10 km	11:00:00 UTC	12:00:00 UTC	1.000
3	Anbieter 3	Angebot	fest	Solingen Hbf	Remscheid Hbf	12:00:00 UTC	13:00:00 UTC	1.000
4	Nachfrager 1	Nachfrage	-	Neuss Hbf	Remscheid Hbf	10:00:00 UTC	13:00:00 UTC	3.000

Tabelle 22: Angebote und Nachfragen im Szenario „Spezialfall 4“

10 Diskussion und Fazit

10.1 Diskussion der Ergebnisse

Die vorliegende Forschungsarbeit zeigt, dass das Konzept einer agentenorientierten Online-Frachtenbörse nicht nur konzipiert, sondern auch in ein erlebbares, lauffähiges Softwaresystem umgesetzt werden kann. Den zunächst konzipierten und anschließend prototypisch implementierten Software-Agenten stehen aus Sicht der technischen Machbarkeit keine Hürden im Weg und es wird gezeigt, dass auch eine über die prototypische Implementierung hinausgehende Entwicklung vorstellbar ist.¹⁴¹⁸

Dieser rein technischen Betrachtungsweise steht jedoch die kritische Betrachtung bzgl. der sozio-ökonomischen Realisierbarkeit der vorgestellten Konzepte gegenüber. Nach Ansicht des Autors sind die größten Hürden bei der praktischen Umsetzung und europaweiten Einführung einer dezentral organisierten Online-Frachtenbörse nicht technischer Natur. Es gilt vielmehr zu eruieren, welche sozialen, wirtschaftlichen und nicht zu letzt auch rechtlichen Rahmenbedingungen in den verschiedenen Märkten und Gesellschaften innerhalb und außerhalb der europäischen Union vorliegen.

Bereits die Technologien Telefax, Btx und Internet stießen im Schienengüterverkehr in der Vergangenheit auf Akzeptanzwiderstände. Es ist anzunehmen, dass eine Branche, die bereits Schwierigkeiten mit der Adaption dieser Technologien hatte, sich weder vorbehaltlos noch enthusiastisch auf eine Technologie einlassen wird, welche einen weiteren wahrgenommenen Kontrollverlust über das eigene Handeln bedeutet. Ferner ist anzunehmen, dass verschiedene Stakeholder im Schienengüterverkehr der Etablierung einer transparenzfördernden Technologie skeptisch bis ablehnend gegenüberstehen werden. Dies betrifft nicht ausschließlich, aber dennoch primär die Eisenbahnverkehrsunternehmen, deren Geschäftsmodell und wirtschaftliche Existenz seit Langem von der Intransparenz des von ihnen bedienten Marktes profitieren.¹⁴¹⁹

Es ist ferner zu erwarten, dass die effizientere Art der Vermittlung von Transportdienstleistungen zwar begrüßt, die hierbei von den Agenten betriebene vollständige Transparenz jedoch auf Kritik stoßen wird. Denn wenngleich die ausgetauschten Daten niemals in der Benutzeroberfläche für Dritte einsehbar sind, so werden – wie in Kapitel 6 beschrieben – sämtliche eingegebenen Daten des menschlichen Benutzers sukzessive über das Netzwerk verteilt.¹⁴²⁰ Der Umstand, dass mit entsprechenden Bemühungen zur Weiterentwicklung der Technologie eine effektive Verschlüsselung der eingegebenen Daten

¹⁴¹⁸ Eine detailliertere Beschreibung dieser Weiterentwicklungspotentiale findet sich in Kapitel 10.2 Vgl. Kapitel 10.2, S. 244.

¹⁴¹⁹ Eine ausführlichere Darstellung der Verhältnisse im Schienengüterverkehrsmarkt findet sich in Kapitel 2.1 Vgl. Kapitel 2.1, S. 7.

¹⁴²⁰ Vgl. Kapitel 6, S. 115.

möglich wird, dürfte hierbei eine in der öffentlichen Wahrnehmung untergeordnete Rolle spielen.

In Bezug auf ein Geschäftsmodell ist zudem anzumerken, dass das vorgestellte Konzept einer autonom funktionierenden Online-Frachtenbörse darauf ausgelegt ist, dass zu ihrem Betrieb keine natürliche oder institutionelle Betreiberperson erforderlich ist. Um einer derartigen Innovation jedoch die zu ihrer Einführung und branchenweiten Anwendung nötige Öffentlichkeit zu verschaffen, wird es sehr wohl einer Betreiberperson bedürfen. Denn wenngleich zum reinen Betrieb der Online-Frachtenbörse kein technischer Betreiber notwendig ist, wird es einer Person bedürfen, welche die Technologie bei den relevanten Stakeholdern bewirbt und sich für die Beratung und Betreuung der ersten Teilnehmer des Marktplatzes verantwortlich zeichnet. Es bleibt zu erforschen, auf welchem Wege diese Person Umsätze generieren kann, um für ihre Arbeit kompensiert zu werden.

Die vorliegende prototypische Implementierung des Konzepts befasst sich zudem nicht mit der Frage, wie das rechtlich bindende Zustandekommen von Transportverträgen aus juristischer Sicht abzusichern wäre.

Aus pragmatischer Sicht stellen sich weitere Fragen zur Implementierung einer autonom und international funktionierenden Online-Frachtenbörse, wie bspw.:

- In welchen Sprachen soll die Benutzeroberfläche verfügbar sein?
- Können Nutzer der Software auch in persönlichen Kontakt miteinander treten, bspw. nach erfolgreichem Handel miteinander?
- Wie könnte dieser Kontakt bei Parteien mit unterschiedlichen Sprachkenntnissen unterstützt werden?

Es wird zu erforschen sein, ob es weitere, bspw. kulturelle Aspekte gibt, welche zu Gestaltungsproblemen bei der Weiterentwicklung der Software führen. Das folgende Kapitel gibt einen Überblick über identifizierte Weiterentwicklungspotentiale.

10.2 Weiterentwicklung der prototypischen Implementierung

10.2.1 Einordnung der Weiterentwicklungspotentiale

Die in dieser Forschungsarbeit vorgestellte Technologie sowie der mit ihrer Hilfe implementierte Softwareprototyp bieten aufgrund ihrer prototypischen Natur und ihres modularen Aufbaus vielfältige Weiterentwicklungspotentiale. Einige dieser Potentiale greifen Ideen zu den Anforderungen an eine Online-Frachtenbörse für den Schienengüterverkehr aus Kapitel 2.4 auf.¹⁴²¹

Eine Beschreibung dieser Potentiale zunächst aus betriebswirtschaftlicher und anschließend aus technologischer Sicht soll diese beleuchten. Es sei jedoch darauf hingewiesen, dass aufgrund der engen Verknüpfung dieser beiden Teilaspekte eine trennscharfe Analyse nach Ansicht des Autors weder möglich noch zweckdienlich ist. Die meisten der aufgeführten Potentiale haben einen betriebswirtschaftlichen wie auch einen technologischen Aspekt.

10.2.2 Weiterentwicklungspotentiale aus betriebswirtschaftlicher Sicht

10.2.2.1 Diskussion der vorliegenden Ergebnisse

Ein grundlegender, erster Schritt bei der Weiterentwicklung der vorliegenden Ergebnisse aus nicht streng technologischer Sicht ist die möglichst breite Veröffentlichung und Diskussion eben dieser Ergebnisse. Ein Dialog zwischen Wissenschaft und Praxis wird als unabdingbar angesehen, um die Chancen und Richtungen möglicher Weiterentwicklungsbemühungen abschätzen zu können.

Während die in dieser Forschungsarbeit entwickelten Ansätze zur Organisation dezentraler Marktplätze einen hohen Innovationsgrad besitzen, ist für die zukünftige Entwicklung entscheidend, ob es gelingt, diese Konzepte in Einklang mit den Vorstellungen von Logistikern zu bringen.

Es wäre hier der Frage nachzugehen, wie ein „realisierbarer Mittelweg“ zwischen jenen den State-of-the-Art bildenden Lösungen zur Frachtvermittlung und den hier präsentierten Konzepten ausgestaltet sein könnte. Ohne einen solchen, zumindest partiellen Abgleich mit der betriebswirtschaftlichen Realität erscheinen weitere Bemühungen zur Weiterentwicklung der Ergebnisse dieser Forschungsarbeit stets dem Vorwurf der „akademischen Realitätsferne“ ausgesetzt.

¹⁴²¹ Es können jedoch auch mit dieser Weiterentwicklung nicht alle, teils widersprüchliche, teils unwirtschaftliche Anforderungen, die im Laufe der Anforderungsanalyse ermittelt wurden, erfüllt werden. Vgl. Kapitel 2.4, S. 14.

10.2.2.2 Tests der Software mit Akteuren aus relevanten Branchen

Aufgrund der beschriebenen Problematik des bisherigen Mangels an Feedback aus der industriellen Praxis könnte ein erster konkreter Schritt hin zu einem Dialog in der Erprobung des Softwareprototyps mit interessierten Akteuren aus bspw. der Logistikbranche oder der Schwerindustrie bestehen.

Die in Kapitel 2.4 beschriebenen Anforderungs-, Umfeld- und Branchenanalysen sowie die finale Implementierung des ORFE-Softwareprototyps haben von derartigen Tests in nicht zu unterschätzender Weise profitiert.¹⁴²²

10.2.2.3 Auktionsmodell

Das verwendete Auktionsmodell ist – wie bereits in Kapitel 7.4.3.4 betont – nicht für das domänenspezifische Problem der Auktion von Transportdienstleistungen angepasst oder ausgelegt.¹⁴²³

Die Literatur zu kombinatorischen Auktionen beschäftigt sich seit längerem mit der Anpassung von Auktionsmodellen an logistische Prozesse, insbesondere in der Beschaffung. Ein auf das Problem dieser Forschungsarbeit (die zweiseitige kombinatorische Auktion von multimodalen Transportdienstleistungen) zugeschnittenes Auktionsmodell ist dem Autor jedoch nicht bekannt.

Es ist wäre durch den modularen Aufbau des Softwareprototyps nicht problematisch, den Ad-hoc-Auktionen ein anderes Auktionsmodell zugrunde zu legen. Daher ist die Konzeption eines adäquateren Auktionsmodells ein wichtiges Weiterentwicklungspotential.

10.2.2.4 Beschränkung auf trusted peers

Ein in Netzwerken verbreitetes Konzept ist das des „peer trust“.¹⁴²⁴

Es wäre eine Weiterentwicklung denkbar, bei dem die Teilnehmer in einem dezentralen Netzwerk anderen Teilnehmern des Netzwerks ihr Vertrauen aussprechen können. Dies könnte bspw. durch die Weitergabe eines Schlüssels in Form einer vertraulichen Datei geschehen. Anschließend könnte die Agenten-Software so modifiziert werden, dass sie nur noch mit Agenten interagiert, denen der Benutzer in dieser Weise das Vertrauen ausgesprochen hat.

¹⁴²² Vgl. Kapitel 2.4, S. 14.

¹⁴²³ Vgl. Kapitel 7.4.3.4, S. 225.

¹⁴²⁴ Vgl. WANG/VARADHARAJAN (2005).

Diese Weiterentwicklung würde dem Wunsch der im Rahmen des Projekts CODE24 befragten Experten nach „geschlossenen Benutzergruppen“ nachkommen.¹⁴²⁵ In der Kategorisierung von Marktplätzen in „offene“ und „geschlossene“ Marktplätze aus Kapitel 5.1.4.3 würden die hierdurch geschaffenen Gruppen „geschlossenen“ Marktplätzen entsprechen.¹⁴²⁶

10.2.2.5 Beschränkung auf web of trust

Ein weitergehender Ansatz wäre das sog. „web of trust“. Das Konzept basiert auf der Annahme, dass der Benutzer eines Agenten anderen Agenten sein Vertrauen aussprechen kann. Hierbei erlaubt bspw. Benutzer A seinem Agenten mit dem Agenten von Benutzer B zu interagieren.

Ein „web of trust“ würde bedeuten, dass der Agent von Benutzer B auch mit allen Agenten interagieren darf, denen Benutzer A ebenfalls sein Vertrauen ausgesprochen hat. Durch diese Weiterentwicklung wäre es wie beim reinen „peer trust“ möglich, dass Unternehmen eigene Marktplätze innerhalb des dezentralen Netzwerks schaffen.

In der Kategorisierung von Marktplätzen in „offene“ und „geschlossene“ Marktplätze aus Kapitel 5.1.4.3 würde dies einem in diesem Kontext neuartigen „halboffenen“ Marktplatz entsprechen.¹⁴²⁷ Der Marktplatz wäre nicht jedem Teilnehmer zugänglich, allerdings müssten einzelne Teilnehmer nicht individuell autorisiert werden, dem Marktplatz beitreten zu können. Dennoch kann über den gezielten Aufbau der Vertrauensbeziehungen eine starke Kontrolle auf die Gruppe ausgeübt werden, bspw. wenn alle relevanten Teilnehmer einem zentralen Akteur vertrauen, sich jedoch nicht gegenseitig. Würde bspw. ein großer Industriekonzern ein solches „web of trust“ um seine logistischen Aktivitäten herum aufbauen, indem die Zugehörigkeit von seinem Vertrauen abhängt, würde der Vertrauensentzug gegenüber einem einzelnen Akteur dessen Ausschluss aus der Gruppe bedeuten.

Hierdurch würde diese Weiterentwicklung dem Wunsch nach „geschlossenen Benutzergruppen“ zumindest in Bezug auf einen eingeschränkten Teilnehmerkreis nachkommen.¹⁴²⁸

10.2.2.6 Zertifikate

Sowohl innerhalb des offenen, dezentralen Marktplatzes als auch innerhalb der soeben skizzierten „halboffenen“ Marktplätze könnte seitens der Praxis eine Vorabprüfung der Teilnehmer gefordert werden, welche die Authentizität und Seriösität der Teilnehmer sicherstellen soll. Eine solche Forderung erscheint wahrscheinlich, da Sicherheits- und

¹⁴²⁵ Vgl. Kapitel 2.4.3, S. 31, sowie Kapitel 2.4.2.3, S. 23.

¹⁴²⁶ Vgl. Kapitel 5.1.4.3, S. 67.

¹⁴²⁷ Vgl. Kapitel 5.1.4.3, S. 67.

¹⁴²⁸ Vgl. Kapitel 2.4.2.1, S. 19.

Authentizitätsprüfungen bei offenen Marktplätzen üblich sind, bestehende multimodale Frachtenbörsen derartige Prüfungen bereits anbieten und sie bereits in der Anforderungsanalyse für den ORFE-Softwareprototyp aufkamen.¹⁴²⁹

Eine Möglichkeit zum Ausschluss nicht ernst gemeinter Angebote und kriminell motivierter Akteure könnte die Vergabe von Zertifikaten darstellen. Eine oder mehrere übergeordnete Organisationen würden diese Zertifikate an Unternehmen verkaufen. Unternehmen können diese Zertifikate gegen Zahlung einer Summe V erwerben, um so ihre Seriösität zu signalisieren. Es sind auch Abstufungen der Zertifizierung denkbar, wie bspw. derart, dass ein Unternehmen länger als einen festzulegenden Zeitraum bestehen oder einen festzulegenden Mindestumsatz nachweisen muss, um eine bestimmte Zertifizierungsstufe zu erhalten.

Die zuvor beschriebene Technik des „peer trust“ könnte dahingehend weiterentwickelt werden, dass Benutzer festlegen können, dass ihre Agenten nur mit solchen Agenten interagieren sollen, die mindestens über die gleiche Zertifizierungsstufe wie sie selbst verfügen.

10.2.2.7 Politisch-rechtliche Potentiale

Es bestehen darüber hinaus eine Vielzahl an Weiterentwicklungspotentialen in Bezug auf die verschiedenen politisch-rechtlichen Rahmenbedingungen, denen sich die Akteure einer tatsächlich international betriebenen Online-Frachtenbörse gegenüber sehen würden.

Es ist eine Vielzahl von Aspekten zu erforschen, welche sich bspw. mit den verwendeten Währungen, Regelungen zum Grenzübertritt bestimmter Frachten, evtl. anfallenden Zöllen sowie unterschiedlichen Sicherheits- und Verfahrensstandards zwischen den verschiedenen von der Online-Frachtenbörse abgedeckten Wirtschaftsregionen betreffen.

10.2.3 Weiterentwicklungspotentiale aus technologischer Sicht

10.2.3.1 Benutzeroberfläche der IDE

Die integrierte Entwicklungsumgebung (IDE) bietet eine grafische Benutzeroberfläche zur Konfiguration von Agenten und Szenarien. Sie stellt eine Alternative zur kommandozeilenbasierten Ausführung derselben Aufgaben dar und wurde entwickelt, um diese Aufgaben auch Benutzern ohne tiefere IT-Kenntnisse zugänglich zu machen.

Dennoch hat die IDE Weiterentwicklungspotentiale bzgl. der Zugänglichkeit insbesondere ihrer Benutzeroberfläche. Eine bessere Benutzerführung insbesondere im Falle fehlerhafter Eingaben könnte hier im Vordergrund stehen, ebenso die stärkere Anleitung des Benutzers

¹⁴²⁹ Vgl. BOTTLE (2015), S. 17, GRIEGER (2003), S. 289, Kapitel 2.4.2.1, S. 19, sowie Kapitel 2.4.3, S. 30.

während seiner ersten Sitzungen und die Implementierung einer interaktiven Hilfefunktion, welche für jede in Kapitel 7.3.3 beschriebene Ansicht verfügbar sein sollte.¹⁴³⁰ Wenn die IDE als Grundlage der in Kapitel 10.2.2.2 angeregten Tests mit Praktikern dienen soll, so muss die Möglichkeit geschaffen werden, dass mehrere Benutzer unabhängig voneinander Projekte erstellen und Simulationen durchführen können.¹⁴³¹

Ein Vorteil der vorliegenden Implementierung besteht in der Tatsache, dass die IDE als internetbasiertes System implementiert ist. Dies ermöglicht es Nutzern ohne IT-Kenntnisse die Software zu erleben, ohne die Software zuvor auf ihren Computer herunterladen und installieren zu müssen. Dennoch ist die Einrichtung eines Internet-Servers zu diesem Zweck immer noch ein Vorgang, welcher ein hohes Maß an IT-Kenntnissen erfordert. Daher kann auch die leichtere Installation dieser Internetserver als Weiterentwicklungspotential identifiziert werden, da so mehr Menschen in die Lage versetzt würden die entwickelte Software zu testen oder zu betreiben.

10.2.3.2 Entwurf einer agenten-individuellen Benutzeroberfläche

Neben diesen Verbesserungen der IDE wäre auch die Implementierung des Prototyps einer agenten-individuellen Benutzeroberfläche denkbar. Dies wäre ein Prototyp der Software, die im Rahmen des AFEX-Konzepts tatsächlich bei Unternehmen zum Einsatz käme und mit Hilfe derer die Konfiguration genau eines Agenten gesteuert würde.

Vor der Konzeption und Implementierung eines solchen Softwareprototyps sollte eine sorgfältige Anforderungsanalyse stehen, welche alle relevanten Eigenschaften einer derartigen Software systematisch ermittelt. Diese Anforderungen dürften aufgrund der Komplexität der zu erfüllenden Aufgabe als auch aufgrund des Innovationsgrades der Software und ihrer Konzepte hoch sein. So wäre bspw. zu ermitteln, wie viele menschliche Benutzer eine Instanz der Software bedienen und welche Funktionen der einzelne Benutzer wahrnehmen würde. Eine Authentifizierung der Benutzer (bspw. durch Eingabe eines Benutzernamens und Passworts) könnte notwendig sein, wenn die Benutzer Rollen wie „Administrator“ und „Händler“ wahrnehmen sollen, welche über verschiedene Benutzerrechte verfügen. So wäre es denkbar, dass im Fall der zuvor beschriebenen Weiterentwicklungspotentiale „peer trust“¹⁴³², „web of trust“¹⁴³³ und „Zertifikatvergabe“¹⁴³⁴ zur Änderung der beschriebenen Schlüssel andere Benutzerrechte erforderlich sind als bspw. zur Änderung der Handelspräferenzen eines Agenten.

¹⁴³⁰ Vgl. Kapitel 7.3.3, S. 191.

¹⁴³¹ Vgl. Kapitel 10.2.2.2, S. 245.

¹⁴³² Vgl. Kapitel 10.2.2.4, S. 245.

¹⁴³³ Vgl. Kapitel 10.2.2.5, S. 246.

¹⁴³⁴ Vgl. Kapitel 10.2.2.6, S. 246.

10.2.3.3 Verschlüsselung der Datenübertragungen

Die Datenübertragung der Agenten-Software findet in der prototypischen Implementierung unverschlüsselt statt. Eine Verschlüsselung sämtlicher übermittelter Daten wäre im Fall einer für den Produktiveinsatz konzipierten Agenten-Software zu gewährleisten.

10.2.3.4 Agentenpräferenzen

Die Agentenpräferenzen bilden die Dimensionen Relation, Zeitfenster, Güterart, Gewicht und Volumen ab. Es wäre zu erforschen, welche weiteren, bspw. qualitativen Dimensionen in den Präferenzen der Agenten erfasst werden könnten.

So könnten bspw. die ALT-Werte, welche zur Zeit aus rein technischen Gründen verwendet werden, um sich gegenseitig ausschließende Angebote in die Auktion einzufügen, dahingehend weiterentwickelt werden, dass Nachfrager mit ihrer Hilfe alternative Routen nachfragen können.¹⁴³⁵ Da Szenarien denkbar erscheinen, in denen bspw. ein Speditionsunternehmen genau eine von zwei Transportdienstleistungen ersteigern möchte, wäre zu erforschen, ob es sinnvoll erscheint, dass Nachfrager entweder eine Route A oder eine Route B nachfragen wollen.

Weiterhin wäre zu erforschen, ob bei der Formulierung von Angebots- und Nachfragepräferenzen seitens der potentiellen Benutzer die Formulierung weiterer Merkmale oder Beziehungen gewünscht ist. So ist es bspw. im vorliegenden Softwareprototyp nicht möglich zu formulieren, dass eine Nachfragepräferenz einen bestimmten Transportmodus bevorzugen oder über einen zuvor festgelegten Haltepunkt als Zwischenstopp laufen soll.

10.2.3.5 Öffnung interner Datenformate für Drittanbieter

Die innerhalb der Agenten-Software eingesetzte domänenspezifische Sprache (DSL) sowie ihre Repräsentation in JSON könnten auch durch Drittanbieter-Software genutzt werden, um eine Anbindung an das AFEX-System zu ermöglichen. Es ist anzumerken, dass diese Öffnung technisch machbar und sinnvoll erscheinen kann, sie jedoch auch Sicherheitsaspekte der bisherigen prototypischen Implementierung betrifft. Wenn im Rahmen einer nicht mehr prototypischen, sondern auf den Produktiveinsatz gerichteten Implementierung Sicherheitsbedenken bzgl. der Verschlüsselung und Weitergabe der sensiblen Daten eine Rolle spielen sollen, wäre dieses Weiterentwicklungspotential kritisch zu evaluieren.

¹⁴³⁵ Vgl. Kapitel 6.4.2, S. 136.

10.2.3.6 Algorithmen und LP-Solver

Die Verwendung einer optimierten, höchst wahrscheinlich kommerziellen Solver-Technologie stellt ein weiteres Weiterentwicklungspotential dar.

Wie in Kapitel 7.4.3.5 beschrieben, wird in der vorliegenden Implementierung ein nicht optimierter Algorithmus zur Lösung des kombinatorischen Problems im Rahmen der Auktionsdurchführung verwendet.¹⁴³⁶

Analog zum Auktionsmodell würde sich auch der zur Lösung der Auktionen genutzte Algorithmus austauschen lassen.

10.2.3.7 Signierte Auktionsergebnisse

Die Signierung digitaler Inhalte zum Zeichen des Konsens zweier Parteien über die weitere Verwendung dieser Inhalte ist elementarer Bestandteil der Internet-Technologie und kommt bspw. bei verschlüsselten E-Mails und E-Commerce-Transaktionen zum Einsatz.

Analog zu diesen Beispielen ließe sich ein Verfahren spezifizieren, mit Hilfe dessen die Agenten in die Lage versetzt werden, sich nicht nur mittels Ad-hoc-Auktion auf eine Allokation zu einigen, sondern diese Einigung Bestandteil eines im Rahmen der Interaktion der Agenten geschlossenen rechtsverbindlichen Vertrages werden zu lassen.

10.3 Weitere Anwendungsgebiete der entwickelten Technologie

10.3.1 Decision-Support-Systeme

Mit den aus dem Prototyp gewonnen Erkenntnissen über zweiseitige kombinatorische Auktionen und verteilte Systeme zur Anbahnung des Handels von elektronisch beschreibbaren Gütern ließen sich Systeme zur Entscheidungshilfe („decision support systems“) konzipieren. Als Auktionsgegenstand wären in diesem Fall neben Transportdienstleistungen auch andere Güter, wie bspw. innerbetriebliche Ressourcen, denkbar. Die Verwendung einseitiger kombinatorischer Auktionen zum innerbetrieblichen Austausch von Logistikdienstleistungen ist bereits in der Vergangenheit diskutiert worden.¹⁴³⁷ Hierbei würde kein „echter“ Handel stattfinden, die Ergebnisse der Auktionen würden jedoch Entscheidern bei der Planung von logistischen Prozessen als Grundlage dienen können.

¹⁴³⁶ Vgl. Kapitel 7.4.3.5, S. 227.

¹⁴³⁷ Vgl. GUJO et al. (2007).

10.3.2 Kontakthanbahnung zwischen potentiellen Geschäftspartnern

Neben den bis hierher vorgestellten Weiterentwicklungspotentialen hin zu mehr Abstraktion der Ergebnisse und Innovation in der Praxis wäre jedoch auch die inhaltlich entgegengesetzte Entwicklungsrichtung denkbar.

So könnte die vorgestellte Technologie auch zum reinen Abgleich („matching“) zwischen Anbieter und Nachfrager verwendet werden. Dies erscheint insbesondere innerhalb klar abgegrenzter Märkte mit hoher Leerfahrtenquote betriebswirtschaftlich interessant. So könnte bspw. innerhalb von Frachthäfen mit Hilfe der Technologie eine dezentrale Kontaktbörse betrieben werden, welche passende Angebote von und Nachfragen nach Transportkapazitäten zusammenbringt. In der Praxis würden hierbei vermutlich keine bindenden Kontrakte geschlossen, sondern lediglich eine Kontaktvermittlung zwischen Anbieter und Nachfrager nach erfolgreichem „matching“ stattfinden.

Aus Sicht dieser Forschungsarbeit würde somit das Konzept des ORFE-Softwareprototyps eine Einsatz- und Weiterentwicklungsmöglichkeit der Technologie des AFEX-Systems darstellen.

10.4 Ausblick auf weitere Forschungsperspektiven

Die aufgezeigten Weiterentwicklungspotentiale bieten eine Vielzahl an interessanten, noch zu erforschenden Fragestellungen.

Die folgende Liste enthält Themenvorschläge für die empirische Forschung zur Akzeptanz und zu den Erfolgsaussichten einer Realisierung der in dieser Forschungsarbeit vorgestellten Konzepte sowie für die kritische Analyse der Erfolgsaussichten ausgewählter Weiterentwicklungspotentiale.

- Ermittlung der Anforderungen einer Clientsoftware für dezentrale Marktplätze,¹⁴³⁸
- Beschreibung eines Systems zum Vertrauensaustausch in dezentralen Netzwerken,¹⁴³⁹
- halboffene elektronische Marktplätze als Teil dezentraler Netzwerke,¹⁴⁴⁰
- Analyse von Zertifikaten als Authentifizierungsmittel in offenen Marktplätzen,¹⁴⁴¹
- kombinatorische Auktionen als Mittel der Entscheidungshilfe: eine bibliographische Analyse,¹⁴⁴²

¹⁴³⁸ Vgl. Kapitel 10.2.3.2, S. 248.

¹⁴³⁹ Vgl. Kapitel 10.2.2.4, S. 245.

¹⁴⁴⁰ Vgl. Kapitel 10.2.2.5, S. 246.

¹⁴⁴¹ Vgl. Kapitel 10.2.2.6, S. 246.

¹⁴⁴² Vgl. Kapitel 10.3.1, S. 250.

- Konzeption eines Auktionsmodells zur Ad-Hoc-Versteigerung von multimodalen Transportdienstleistungen¹⁴⁴³ sowie
- Konzeption eines Systems zur verschlüsselten Signierung der Auktionsergebnisse.¹⁴⁴⁴

Aus pragmatischer Sicht erscheint jedoch die weitere Diskussion des Konzepts einer „agentenorientierten Frachtenbörse“ erforderlich zu sein, so dass sich die kurzfristige Forschung auf den Dialog zwischen Wissenschaft und Praxis konzentrieren sollte.

10.5 Fazit

Im ersten Teil dieser Forschungsarbeit wurden Konzepte zu autonomen, dezentral organisierten Software-Agenten zur Durchführung zweiseitiger kombinatorischer Auktionen von multimodalen Transportdienstleistungen am Beispiel des kombinierten Verkehrs erarbeitet. Dies geschah auf Basis der Zusammenführung von Erkenntnissen über die Erfordernisse einer Online-Frachtenbörse für Transportdienstleistungen im Schienengüterverkehr, über den Einsatz zweiseitiger kombinatorischer Auktionen sowie über die Gestaltung autonomer Multi-Agenten-Systeme und konstuierte das erste wissenschaftliche Problem dieser Forschungsarbeit.

Wie gezeigt wurde, ist der entwickelte Softwareprototyp eine vollfunktionsfähige Software zur Erprobung der in dieser Forschungsarbeit vorgestellten Konzepte. Die Software bietet durch ihre modulare Programmierweise und technische Infrastruktur ein solides Fundament für die weitere Entwicklung und kann der weiteren Forschung und einer möglichen Reimplementierung in vielen Bereichen als Vorlage und Anschauungsbeispiel dienen. Dies konstituierte das zweite wissenschaftliche Problem dieser Forschungsarbeit.

Dennoch ist der vorliegende Softwareprototyp nur ein Prototyp. Er ist in weiten Teilen als eine Machbarkeitsstudie zu den im ersten Teil der Forschungsarbeit vorgestellten Konzepten zu sehen und mit anderen Prioritäten implementiert worden als eine professionelle, auf den Produktiveinsatz hin entwickelte Software. Er bildet ein gesamtheitlich durchdachtes System flexibler Strukturen und wohldefinierter Schnittstellen, welches der weiteren angewandten Forschung und betrieblichen Evaluierung dienen soll.

Die prototypische Implementierung des AFEX-Systems als zweites wissenschaftliches Problem dieser Forschungsarbeit diene damit – wie andere Machbarkeitsstudien auch – zuerst dem Anspruch, den Dialog zwischen verschiedenen Interessengruppen zu fördern und speziell auch nach Abschluss des Projekts „CODE24“ einen Dialog über mögliche zukünftige Entwicklungen hinsichtlich der Vermittlung von multimodalen Transportdienstleistungen am Beispiel des Kombinierten Verkehrs zu ermöglichen.

¹⁴⁴³ Vgl. Kapitel 10.2.2.3, S. 245.

¹⁴⁴⁴ Vgl. Kapitel 10.2.3.7, S. 250.

Literaturverzeichnis

Alle Quellen werden nach folgendem Schema aufgeführt:

- In der ersten Zeile wird der Referenztitel der Quelle angegeben. Wird im Text auf eine Quelle referenziert, entspricht die Quellenangabe dieser Form.
- Wurde das Werk von einem Autoren verfasst, entspricht der Referenztitel dessen Nachnamen, gefolgt von dem Erscheinungsjahr der Quelle in Klammern.
- Wurde das Werk von zwei Autoren verfasst, werden die Nachnamen beider Autoren aufgeführt (getrennt von einem Schrägstrich).
- Wurde das Werk von mehr als zwei Autoren verfasst, wird der Nachname des ersten Autors mit dem Zusatz „et al.“ aufgeführt.
- Zu Online-Quellen wird die für die Veröffentlichung verantwortliche Instanz (juristische oder natürliche Person) aufgeführt.
- Zu allen Online-Quellen wird das Datum des letzten Zugriffs sowie die zu diesem Zeitpunkt gültige Internetadresse (URL) angegeben.
- Die im Referenztitel von Online-Quellen verwendete Jahresangabe entspricht dem Veröffentlichungsdatum des Inhalts unter der URL.
- Im Falle der Startseite von Internetpräsenzen von Unternehmen, Vereinen oder Projekten wird das Datum des letzten Zugriffs als Veröffentlichungsdatum angenommen, da nicht nachvollziehbar ist, wann die Quelle in ihrer vorliegenden Form veröffentlicht wurde.

Ackermann et al. (2011)

Ackermann, H.; Ewe, H.; Kopfer, H.; Küfer, K. H.: Combinatorial Auctions in Freight Logistics. In: Computational Logistics, Second International Conference, ICCL Proceedings 2011, 19.-22.09.2011 in Hamburg, Springer, Berlin - Heidelberg, 2011, S. 1–17.

Ammelburger/Scherer (2008)

Ammelburger, D.; Scherer, R.: Webentwicklung mit CakePHP. 1. Auflage, O'Reilly, Sebastopol, 2008.

Anderson et al. (2010)

Anderson, J.; Lehnardt, J.; Slater, N.: CouchDB: The Definitive Guide. 1. Auflage, O'Reilly, Sebastopol, 2010.

Andrew/Shafer (2006)

Andrew, R.; Shafer, D.: CSS – Anspruchsvolle Websites mit Cascading Stylesheets – Grundlagen, Designtechniken und Referenz. 2. Auflage, dpunkt.verlag, Heidelberg, 2006.

Apache (2017)

o.V.: Internetpräsenz unter der Domain www.apache.org. Online-Quelle unter der URL: <http://www.apache.org>, Datum des Zugriffs: 07.07.2017.

Ausubel/Cramton (1998)

Ausubel, L. M.; Cramton, P. C.: The optimality of being efficient. World Bank, Private Sector Development Department, Private Participation in Infrastructure Division, 1998.

Bazaraa et al. (2010)

Bazaraa, M. S.; Jarvis, J. J.; Sherali, H. D.: Linear programming and network flows. John Wiley and Sons, New York, 2010.

Bedrouni et al. (2009)

Bedrouni, A.; Mittu, R.; Boukhtouta, A.; Berger, J.: Distributed Intelligent Systems. Springer, Berlin - Heidelberg, 2009.

Berlin/Rooney (2006)

Berlin, D.; Rooney, G.: Practical Subversion. 2. Auflage, Apress, Berkeley, 2006.

Beydoun et al. (2009)

Beydoun, G.; Low, G.; Henderson-Sellers, B.; Mouratidis, H.; Gomez-Sanz, J.; Pavón, J.; Gonzalez-Perez, C.: FAML: A Generic Metamodel for MAS Development.

Bichler et al. (2005)

Bichler, M.; Pikovsky, A.; Setzer, T.: Kombinatorische Auktionen in der betrieblichen Beschaffung – Eine Analyse grundlegender Probleme. In: Wirtschaftsinformatik, Jg. 47 (2005) Nr. 2, S. 126–134.

Bing Maps (2017)

Bing Maps: Internetpräsenz unter der Domain www.bing.com. Online-Quelle unter der URL: <http://www.bing.com/maps/>, Datum des Zugriffs: 07.07.2017.

Bordini et al. (2006)

Bordini, R. H.; Braubach, L.; Dastani, M.; Seghrouchni, A.; Gomez-Sanz, J.; Leite, J.; O'Hare, G.; Pokahr, A.; Ricci, R.: A Survey of Programming Languages and Platforms for Multi-Agent Systems. In: Informatica, Jg. 30 (2006) Nr. 1, S. 33–44.

Bottler (2015)

Bottler, St.: Serviceoffensive der Börsen. In: verkehrsRUNDschau – Wochenmagazin für Spedition, Transport und Logistik, Jg. 41 (2015), S. 16–17.

Bray et al. (2006)

Bray, T.; Paoli, J.; Sperberg-McQueen, C. M.; Maler, E.; Yergeau, F.; Cowan, J.: Extensible Markup Language (XML) – 1.1 (Second Edition). Online-Quelle unter der URL: <http://www.w3pdf.com/W3cSpec/XML/2/REC-xml11-20060816.pdf>, Datum des Zugriffs: 10.06.2013.

Bresciani et al. (2002)

Bresciani, P.; Giorgini, P.; Giunchiglia, F.; Mylopoulos, J.; Perini, A.: Tropos: An agent-oriented software development methodology. In: Autonomous Agents and Multi-Agent Systems, Jg. 8 (2002) Nr. 3, S. 203–236.

Bruns et al. (2010a)

Bruns, A. S.; Günes, N.; Zelewski, S.: Online-Frachtenbörse für den transeuropäischen Schienengüterverkehr. In: Internationales Verkehrswesen, Jg. 62 (2010) Nr. 11, S. 25–29.

Bruns et al. (2010b)

Bruns, A. S.; Zelewski, S.; Klumpp, M.: Trends in der Güterverkehrslogistik – eine Expertenbefragung unter besonderer Berücksichtigung von Schienengüterverkehren. Projektbericht des Verbundprojekts CODE24 Nr. 1, Institut für Produktion und Industrielles Informationsmanagement, Universität Duisburg-Essen, Campus Essen, Essen, 2010.

Bruns et al. (2012a)

Bruns, A. S.; Zelewski, S.; Lehr, T.: Analyse des Marktpotenzials für Online-Frachtenbörsen im Segment von Schienengüterverkehren und Kombinierten Güterverkehren. Projektbericht des Verbundprojekts CODE24 Nr. 6, Institut für Produktion und Industrielles Informationsmanagement, Universität Duisburg-Essen, Campus Essen, Essen, 2012.

Bruns et al. (2012b)

Bruns, A. S.; Föhring, R.; Zelewski, S.; Günes, N.: Eine Online-Frachtenbörse für Verlader und Güterbahnen. In: Güterbahnen – Güterverkehr auf der Schiene: Markt • Technik • Verkehrspolitik, Jg. 11 (2012) Nr. 1, S. 33–37.

Burmeister et al. (1997)

Burmeister, B.; Haddadi, A.; Matylis, G.: Application of multi-agent systems in traffic and transportation. In: IEE Proceedings on Software Engineering, Jg. 144 (1997) Nr. 1, S. 51–60.

Buttermann (2002)

Buttermann, V.: Strategische Allianzen im europäischen Eisenbahngüterverkehr. Dissertation, Dresden, 2002.

Carlson/Richardson (2006)

Carlson, L.; Richardson, L.: Ruby Cookbook. O'Reilly, Sebastopol, 2006.

Carneiro/Barazi (2010)

Carneiro, C.; Barazi, R.: Beginning Rails 3. 1. Auflage, Apress, Berkeley, 2010.

Casaló et al. (2008)

Casaló, L. V.; Flavián, C.; Guinalíu, M.: The role of satisfaction and website usability in developing customer loyalty and positive word-of-mouth in the e-banking services. In: International Journal of Bank Marketing, Jg. 26 (2008) Nr. 6, S. 399–417.

Chacon (2014)

Chacon, S.: Pro Git. 2. Auflage, Apress, Berkeley, 2014.

Chak (2009)

Chak, D.: Enterprise Rails. 1. Auflage, O'Reilly, Sebastopol, 2009.

Chang et al. (2009)

Chang, R.-S.; Chang, J.-S.; Lin, P.-S.: An ant algorithm for balanced job scheduling in grids. In: Future Generation Computer Systems, Jg. 25 (2009) Nr. 1, S. 20–27.

CODE24 (2017)

CODE24: Internetpräsenz unter der Domain egtc-rhine-alpine.eu. Online-Quelle unter der URL: <http://egtc-rhine-alpine.eu/code24/>, Datum des Zugriffs: 07.07.2017.

Cooper (2009)

Cooper, P.: Beginning Ruby: From Novice to Professional. 2. Auflage, Apress, Berkeley, 2009.

Cossentino et al. (2010)

Cossentino, M.; Gaud, N.; Hilaire, V.; Galland, S.; Koukam, A.: ASPECS: an agent-oriented software process for engineering complex systems. In: Autonomous Agents and Multi-Agent Systems, Springer, Berlin - Heidelberg, Jg. 20 (2010) Nr. 2, S. 260–304.

CouchdbApache (2017)

o.V.: Internetpräsenz unter der Domain couchdb.apache.org. Online-Quelle unter der URL: <http://couchdb.apache.org>, Datum des Zugriffs: 07.07.2017.

Crainic et al. (2006)

Crainic, T. G.; Le Cun, B.; Roucairol, C.: Parallel branch-and-bound algorithms. In: Parallel combinatorial optimization, 2006, S. 1–28.

Crockford (2006)

Crockford, D.: JSON – The x in Ajax. Online-Quelle unter der URL: <http://www.json.org/json.pdf>, Datum des Zugriffs: 10.06.2013.

Crockford (2008)

Crockford, D.: JavaScript: The Good Parts – Unearthing the Excellence in JavaScript. 1. Auflage, O'Reilly, Sebastopol, 2008.

Davidsson et al. (2005)

Davidsson, P.; Henesey, L.; Ramstedt, L.; Törnquist, J.; Wernstedt, F.: An analysis of agent-based approaches to transport logistics. In: Transportation Research Part C: Emerging Technologies, Elsevier, Amsterdam, Jg. 13 (2005) Nr. 4, S. 255–271.

DB Cargo (2017)

DB Cargo: Internetpräsenz unter der Domain www.dbcargo.com. Online-Quelle unter der URL: http://www.dbcargo.com/rail-deutschland-de/products_services/eservices/freight_schedule.html, Datum des Zugriffs: 07.07.2017.

Deering/Hinden (1998)

Deering, S.; Hinden, R.: Internet Protocol, Version 6 (IPv6) Specification. Online-Quelle unter der URL: <https://tools.ietf.org/html/rfc2460>, Datum des Zugriffs: 01.01.2016.

DIUM (2017)

DIUM: Internetpräsenz unter der Domain dium.dbschenker.com. Online-Quelle unter der URL: <http://dium.dbschenker.com/dium/index.jsp>, Datum des Zugriffs: 07.07.2017.

Dörr/Endemann (2014)

Dörr, H.; Endemann, P.: Moving Forward Freight Mobility Innovations. In: Internationales Verkehrswesen, Jg. 66 (2014) Nr. 2, S. 60–63.

Ediger (2008)

Ediger, B.: Advanced Rails. 1. Auflage, O'Reilly, Sebastopol, 2008.

Endemann/Kaspers (2011)

Endemann, P.; Kaspers, Th.: Online-Transportbörse soll Schienengüterverkehr stärken – Was braucht die Praxis? In: Güterbahnen – Güterverkehr auf der Schiene: Markt • Technik • Verkehrspolitik, Jg. 10 (2011) Nr. 4, S. 31–36.

Erdweg et al. (2014)

Erdweg, S.; Fehrenbach, St.; Ostermann, K.: Evolution of software systems with extensible languages and DSLs. In: Software, IEEE, Jg. 31 (2014) Nr. 5, S. 68–75.

Ferber et al. (2004)

Ferber, J.; Gutknecht, O.; Michel, F.: From agents to organizations: An organizational view of multi-agent systems. In: Agent-Oriented Software Engineering IV, Springer, Berlin - Heidelberg, 2004, S. 214–230.

Fidan/Kuhlmann (2013)

Fidan, S.; Kuhlmann, A. S.: Branchenanalyse für die Vermittlung von Gütertransportdienstleistungen mit Hilfe von Online-Frachtenbörsen im Bereich des europäischen Verkehrskorridors 24. Projektbericht des Verbundprojekts CODE24 Nr. 8, Institut für Produktion und Industrielles Informationsmanagement, Universität Duisburg-Essen, Campus Essen, Essen, 2013.

FIPA (2017)

FIPA: Internetpräsenz unter der Domain www.fipa.org. Online-Quelle unter der URL: <http://www.fipa.org>, Datum des Zugriffs: 07.07.2017.

Fischer (2008)

Fischer, T.: Geschäftsmodelle in den Transportketten des europäischen Schienengüterverkehrs. Eine Typologisierung von Eisenbahnverkehrsunternehmen unter besonderer Berücksichtigung der Anbieterstruktur im deutschsprachigen Raum. Dissertation, Wien, 2008.

Flanagan (2011)

Flanagan, D.: JavaScript – The Definitive Guide. 6. Auflage, O'Reilly, Sebastopol, 2011.

Flanagan/Matsumoto (2008)

Flanagan, D.; Matsumoto, Y.: The Ruby Programming Language. O'Reilly, Sebastopol, 2008.

Fortino/Russo (2012)

Fortino, G.; Russo, W.: ELDAMeth: An agent-oriented methodology for simulation-based prototyping of distributed agent systems. In: Information and Software Technology, Jg. 54 (2012) Nr. 6, S. 608–624.

Fox et al. (2000)

Fox, M. S.; Barbuceanu, M.; Teigen, R.: Agent-oriented supply-chain management. In: International Journal of Flexible Manufacturing Systems, Jg. 12 (2000) Nr. 2-3, S. 165–188.

Franklin/Graesser (1997)

Franklin, S.; Graesser, A.: Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents. In: Intelligent Agents III – Agent Theories, Architectures, and Languages, Springer, Berlin - Heidelberg, 1997, S. 21–35.

FreitOne (2017)

o.V.: Internetpräsenz unter der Domain www.freit-one.de. Online-Quelle unter der URL: <http://www.freit-one.de>, Datum des Zugriffs: 07.07.2017.

Friedrich (2009)

Friedrich, M.: Anwendungsdomänen für Mobile Agenten. Dissertation, Tübingen, 2009.

Föhring et al. (2012)

Föhring, R.; Kuhlmann, A. S.; Zelewski, S.: Vorstellung des finalen Softwareprototyps einer Online-Frachtenbörse. Projektbericht des Verbundprojekts CODE24 Nr. 7, Institut für Produktion und Industrielles Informationsmanagement, Universität Duisburg-Essen, Campus Essen, Essen, 2012.

Föhring/Bruns (2011)

Föhring, R.; Bruns, A. S.: Implementierung des Software-Prototyps einer Online-Frachtenbörse. Projektbericht des Verbundprojekts CODE24 Nr. 3, Institut für Produktion und Industrielles Informationsmanagement, Universität Duisburg-Essen, Campus Essen, Essen, 2011.

Föhring/Zelewski (2013)

Föhring, R.; Zelewski, S.: ORFE: Online Rail Freight Exchange – a software prototype for the configuration of multi-modal supply chains focussed on rail freight traffic along the Alps Transversal Rotterdam-Genoa. OrGoLo-Projektbericht Nr. 20, Institut für Produktion und Industrielles Informationsmanagement, Universität Duisburg-Essen, Campus Essen, Essen, 2013.

Föhring/Zelewski (2015a)

Föhring, R.; Zelewski, S.: ORFE to AFEX - A Conceptual Look into the Future of Online Freight Exchanges. In: Clausen, U., Friedrich, H., Thaller, C., Geiger, C. (Hrsg.): Commercial Transport, Proceedings of the 2nd Interdisciplinary Conference on Production Logistics and Traffic, 21.-22.07.2015 in Dortmund, Springer, Berlin - Heidelberg, 2015, S. 241–253.

Föhring/Zelewski (2015b)

Föhring, R.; Zelewski, S.: Intelligent exchanges and coordination in multimodal supply chains. In: Blecker, Th., Kersten, K., Ch. M. Ringle (Hrsg.): Operational Excellence in Logistics and Supply Chains, Berlin, 2015, S. 355–377.

Föhring/Zelewski (2015c)

Föhring, R.; Zelewski, S.: Coordination in multimodal Supply Chains: Drafting an agent-based exchange. In: Mattfeld, D., Spengler, Th., Brinkmann, J., Grunewald, M. (Hrsg.): Lecture Notes in Logistics. Contributions of the Section Logistics of the German Academic Association for Business Research, Springer, Berlin - Heidelberg, 2015, S. 63–74.

Föhring/Zelewski (2015d)

Föhring, R.; Zelewski, S.: Towards Decentralized Electronic Market Places and Agent-Based Freight Exchanges for Multimodal Transports. In: Proceedings of the IEEE International Conference on Automation Science and Engineering (CASE), Amsterdam, 2015, S. 249–254.

Föhring/Zelewski (2015e)

Föhring, R.; Zelewski, S.: AFEX: An autonomous freight exchange concept. In: Santos, B. F., Correia, G. H. A. and Kroesen, M. (Hrsg.): Transportation Research Procedia, Amsterdam, 2015, S. 644–651.

Gamble et al. (2010)

Gamble, A.; Cloves, C.; Barazi, R.: Beginning Rails 3. 1. Apress, Berkeley, 2010.

Gearhart et al. (2013)

Gearhart, J. L.; Adair, K. L.; Detry, R. J.; Durfee, J. D.; Jones, K. A.; Martin, N.: Comparison of Open-Source Linear Programming Solvers. Online-Quelle unter der URL: <http://prod.sandia.gov/techlib/access-control.cgi/2013/138847.pdf>, Datum des Zugriffs: 01.05.2016.

GitSCM (2017)

GitSCM: Internetpräsenz unter der Domain git-scm.com. Online-Quelle unter der URL: <http://git-scm.com>, Datum des Zugriffs: 07.07.2017.

GNU (2017)

GNU: Internetpräsenz unter der Domain www.gnu.org. Online-Quelle unter der URL: <http://www.gnu.org/software/glpk/>, Datum des Zugriffs: 07.07.2017.

Google (2017)

Google: Internetpräsenz unter der Domain maps.google.com. Online-Quelle unter der URL: <https://maps.google.com>, Datum des Zugriffs: 07.07.2017.

Gouaïch/Michel (2005)

Gouaïch, A.; Michel, F.: Towards a Unified View of the Environment(s) within Multi-Agent Systems. In: *Informatica*, Jg. 29 (2005) Nr. 4, S. 423–432.

Grieger (2003)

Grieger, M.: Electronic marketplaces: A literature review and a call for supply chain management research. In: *European Journal of Operational Research*, Jg. 144 (2003), S. 280–294.

Gujo et al. (2007)

Gujo, O.; Schwind, M.; Vykoukal, J.; Weiß, K.: ComEx: Kombinatorische Auktionen zum innerbetrieblichen Austausch von Logistikdienstleistungen. Online-Quelle unter der URL: <http://www.is-frankfurt.de/publikationenNeu/ComExKombinatorischeAuktionen1957.pdf>, Datum des Zugriffs: 26.06.2013.

Habib/Bruns (2012)

Habib, S.; Bruns, A. S.: Analyse des kompetitiven Umfelds für die Etablierung einer Online-Frachtenbörse zur Vermittlung von Eisenbahngüterverkehren auf der Nord-Süd-Transversale „Rotterdam-Genua“. Projektbericht des Verbundprojekts CODE24 Nr. 5, Institut für Produktion und Industrielles Informationsmanagement, Universität Duisburg-Essen, Campus Essen, Essen, 2012.

HaCon (2017)

HaCon: Internetpräsenz unter der Domain gueterfahrplan.hacon.de. Online-Quelle unter der URL: <http://gueterfahrplan.hacon.de/bin/db/query.exe/dn>, Datum des Zugriffs: 07.07.2017.

Haklay (2010)

Haklay, M.: How good is volunteered geographical information? A comparative study of OpenStreetMap and Ordnance Survey datasets. In: *Environment and Planning. B, Planning & design*, Jg. 37 (2010) Nr. 4, S. 682–703.

Haklay/Weber (2008)

Haklay, M.; Weber, P.: Openstreetmap: User-generated street maps. In: Pervasive Computing, Jg. 7 (2008) Nr. 4, S. 12–18.

Harold/Means (2009)

Harold, E. R.; Means, W. S.: XML in a nutshell. 3. Auflage, O'Reilly, Sebastopol, 2009.

Ickert et al. (2007)

Ickert, L.; Matthes, U.; Rommerskirchen, S.; Weyand, E.; Schlesinger, M.; Limbers, J.: Abschätzung der langfristigen Entwicklung des Güterverkehrs in Deutschland bis 2050. Schlussbericht der ProgTrans AG vom 31.05.2007 im Auftrag des Bundesministeriums für Verkehr, Bau und Stadtentwicklung. Basel, 2007. Auch im Internet verfügbar unter der URL: <http://www.bmvbs.de/cae/servlet/contentblob/30886/publicationFile/455/gueterverkehrs-prognose-2050.pdf>, Datum des Zugriffs: 21.02.2012.

Jennings (2000)

Jennings, N. R.: On agent-based software engineering. In: Artificial Intelligence, Jg. 117 (2000) Nr. 2, S. 277–296.

Jennings (2001)

Jennings, N. R.: An Agent-based Approach For Building Complex Software Systems. In: Communications of the ACM, Jg. 44 (2001) Nr. 4, S. 35–41.

Json (2017)

o.V.: Internetpräsenz unter der Domain www.json.org. Online-Quelle unter der URL: <http://www.json.org>, Datum des Zugriffs: 07.07.2017.

Kasperski/Kulej (2009)

Kasperski, A.; Kulej, M.: Choosing robust solutions in discrete optimization problems with fuzzy costs. In: Fuzzy Sets and Systems, Jg. 160 (2009) Nr. 5, S. 667–682.

Kay (1993)

Kay, A. C.: The Early History Of Smalltalk. Online-Quelle unter der URL: <http://esug.org/data/HistoricalDocuments/Smalltalk80/SmalltalkHistory.pdf>, Datum des Zugriffs: 27.05.2013.

Kay (1998)

Kay, A. C.: Alan Kay On Messaging. Online-Quelle unter der URL: <http://c2.com/cgi/wiki?AlanKayOnMessaging>, Datum des Zugriffs: 27.05.2013.

Kay (2003)

Kay, M. H.: XML Five Years On: A Review of the Achievements So Far and the Challenges Ahead. Online-Quelle unter der URL: <http://www.saxonica.com/papers/DocEng2003.pdf>, Datum des Zugriffs: 02.07.2013.

Kersken (2007)

Kersken, S.: Praxiswissen Ruby. 1. Auflage, O'Reilly Verlag GmbH & Co. KG, Köln, 2007.

Klippert et al. (2011)

Klippert, S. R.; Kowalski, M.; Bruns, A. S.: Anforderungsanalyse für eine Online-Frachtenbörse im Eisenbahngüterverkehr – Entwicklung einer Anforderungsspezifikation aus betriebswirtschaftlicher Perspektive. Projektbericht des Verbundprojekts CODE24 Nr. 2, Institut für Produktion und Industrielles Informationsmanagement, Universität Duisburg-Essen, Campus Essen, Essen, 2011.

Klippert et al. (2013)

Klippert, S. R.; Zelewski, S.; Kuhlmann, A. S.; Kowalski, M.: Schienentransporte im KV über eine Online-Frachtenbörse organisieren – Das EU-Verbundprojekt Code24 gibt neue Impulse für den Kombinierten Verkehr (KV) auf der Schiene. In: EI – Der Eisenbahningenieur, Jg. 64 (2013) Nr. 5, S. 91–96.

Koboldt et al. (2003)

Koboldt, C.; Maldoom, D.; Marsden, R.: The First Combinatorial Spectrum Auction – Lessons from the Nigerian auction of fixed wireless access licences. dotecon, London, 2003.

Kofler/Kramer (2005)

Kofler, M.; Kramer, D.: The Definitive Guide to MySQL 5. 1. Auflage, Apress, Berkeley, 2005.

Kollmann (2001)

Kollmann, T.: Virtuelle Marktplätze: Grundlagen, Management, Fallstudie. Vahlen, München, 2001.

Kollmann (2016)

Kollmann, T.: E-Business: Grundlagen elektronischer Geschäftsprozesse in der Digitalen Wirtschaft. 6. Auflage, Springer Gabler, Wiesbaden, 2016.

Kolovos et al. (2006)

Kolovos, D. S.; Paige, R. F.; Kelly, T.; Polack, F. A. C.: Requirements for Domain-Specific Languages. Online-Quelle unter der URL: http://www-users.cs.york.ac.uk/~tpk/req_dsls.pdf, Datum des Zugriffs: 10.06.2013.

Kreibich (2010)

Kreibich, J.: Using SQLite. 1. Auflage, O'Reilly, Sebastopol, 2010.

Krishna (2009)

Krishna, V.: Auction Theory. 2. Auflage, Academic Press, San Diego, 2009.

Krishna/Perry (1998)

Krishna, V.; Perry, M.: Efficient mechanism design. Online-Quelle unter der URL: <http://econ.la.psu.edu/~vkrishna/papers/vcg20.pdf>, Datum des Zugriffs: 19.06.2013.

König/Jugelt (2009)

König, R.; Jugelt, R.: Neue Wege für die Einbindung des Schienengüterverkehrs in die Wertschöpfungsketten der Logistik. In: Wissenschaftliche Zeitschrift der Technischen Universität Dresden, Jg. 58 (2009) Nr. 1-2, S. 115–120.

Langtangen (2009)

Langtangen, H. P.: Python Scripting for Computational Science. 3. Auflage, Springer, Berlin - Heidelberg, 2009.

Laurie/Laurie (2003)

Laurie, B.; Laurie, P.: Apache – Das umfassende Handbuch. 2. Auflage, O'Reilly, Sebastopol, 2003.

Leach et al. (2005)

Leach, P. J.; Mealling, M.; Salz, R.: A Universally Unique Identifier (UUID) URN Namespace. Online-Quelle unter der URL: <https://tools.ietf.org/html/rfc4122>, Datum des Zugriffs: 01.01.2016.

Leal-Taixé et al. (2011)

Leal-Taixé, L.; Pons-Moll, G.; Rosenhahn, B.: Everybody needs somebody: Modeling social and grouping behavior on a linear programming multiple people tracker. In: ICCV Workshop on Modeling, Simulation and Visual Analysis of Large Crowds, 2011, S. 120–127.

Lee (2011)

Lee, D. A.: JXON: an architecture for schema and annotation driven JSON/XML bidirectional transformations. In: Proceedings of Balisage: The Markup Conference 2011. Balisage Series on Markup Technologies, Jg. 7 (2011), Online-Quelle unter der URL: <http://www.balisage.net/Proceedings/vol7/html/Lee01/BalisageVol7-Lee01.html>, Datum des Zugriffs: 02.07.2013.

Lennon (2009)

Lennon, J.: Beginning CouchDB. 1. Auflage, Apress, Berkeley, 2009.

Lerdorf (1995)

Lerdorf, R.: Announce: Personal Home Page Tools (PHP Tools). Newsgroup: comp.infosystems.www.authoring.cgi, 1995. Auch im Internet verfügbar unter der URL: <https://groups.google.com/forum/#!msg/comp.infosystems.www.authoring.cgi/PyJ25gZ6z7A/M9FkTUVdfcwJ>, Datum des Zugriffs: 01.01.2016.

Liberty/MacDonald (2009)

Liberty, J.; MacDonald, B.: Learning C# 3.0. O'Reilly, Sebastopol, 2009.

Lippert (2004)

Lippert, E.: Rumours of VBScript's Death Have Been Greatly Exaggerated. Online-Quelle unter der URL: <http://blogs.msdn.com/b/ericlippert/archive/2004/04/09/110508.aspx>, Datum des Zugriffs: 21.09.2010.

Loeliger/McCullough (2012)

Loeliger, J.; McCullough, M.: Version Control with Git. 2. Auflage, O'Reilly, Sebastopol, 2012.

Malone/Crowston (1994)

Malone, T. W.; Crowston, K.: The Interdisciplinary Study of Coordination. In: ACM Computing Surveys (CSUR), Jg. 26 (1994) Nr. 1, S. 87–119.

Martelli (2006)

Martelli, A.: Python in a Nutshell. 2. Auflage, O'Reilly, Sebastopol, 2006.

McAfee/McMillan (1987)

McAfee, R. P.; McMillan: Auctions and Bidding. In: Journal of Economic Literature, Jg. 25 (1987) Nr. 2, S. 699–738.

Meindl/Templ (2012)

Meindl, B.; Templ, M.: Analysis of commercial and free and open source solvers for linear optimization problems. Online-Quelle unter der URL: http://neon.vb.cbs.nl/cascprivate/..%5Ccasc%5CESSNet2%5Cdeliverable_solverstudy.pdf, Datum des Zugriffs: 12.09.2015.

Meyer (2007)

Meyer, E.: CSS – The Definitive Guide. 3. Auflage, O'Reilly, Sebastopol, 2007.

Microsoft (2017)

Microsoft: Internetpräsenz unter der Domain www.microsoft.com. Online-Quelle unter der URL: <http://www.microsoft.com/express/>, Datum des Zugriffs: 07.07.2017.

Milgrom/Weber (1982)

Milgrom, P. R.; Weber, R. J.: A Theory of Auctions and Competitive Bidding. In: *Econometrica*, Jg. 50 (1982) Nr. 5, S. 1089–1122.

Morsy/Otto (2008)

Morsy, H.; Otto, T.: *Ruby on Rails 2 – Das Entwickler-Handbuch*. 1. Auflage, Galileo Computing, Bonn, 2008.

Musciano/Kennedy (2007)

Musciano, C.; Kennedy, B.: *HTML and XHTML – The Definitive Guide*. 6. Auflage, O'Reilly, Sebastopol, 2007.

MySQL (2017)

MySQL: Internetpräsenz unter der Domain mysql.com. Online-Quelle unter der URL: <http://mysql.com>, Datum des Zugriffs: 07.07.2017.

Nash (2010)

Nash, T.: *Accelerated C# 2010*. Apress, Berkeley, 2010.

Neis et al. (2011)

Neis, P.; Zielstra, D.; Zipf, A.: The street network evolution of crowdsourced maps: OpenStreetMap in Germany 2007–2011. In: *Future Internet*, Jg. 4 (2011) Nr. 1, S. 1–21.

Niemeyer/Knudsen (2005)

Niemeyer, P.; Knudsen, J.: *Learning Java*. 3. Auflage, O'Reilly, Sebastopol, 2005.

Nikolai/Madey (2009)

Nikolai, C.; Madey, G.: Tools of the Trade: A Survey of Various Agent Based Modeling Platforms. In: *Journal of Artificial Societies and Social Simulation*, Jg. 12 (2009) Nr. 2, o.S.

Oki (2012)

Oki, E.: *Linear programming and algorithms for communication networks: a practical guide to network design, control, and management*. CRC Press, Boca Raton, 2012.

Omicini et al. (2004)

Omicini, A.; Ricci, A.; Viroli, M.; Castelfranchi, C.; Tummolini, L.: Coordination artifacts: Environment-based coordination for intelligent agents. In: Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems, 19.-23.07.2004 in New York, Bd. 1, IEEE Computer Society, 2004, S. 286–293.

OpenDataCommons (2017)

OpenDataCommons: Internetpräsenz unter der Domain opendatacommons.org. Online-Quelle unter der URL: <http://opendatacommons.org/licenses/odbl/>, Datum des Zugriffs: 07.07.2017.

Openstreetmap (2017)

o.V.: Internetpräsenz unter der Domain www.openstreetmap.org. Online-Quelle unter der URL: <http://www.openstreetmap.org>, Datum des Zugriffs: 07.07.2017.

Oracle (2017)

Oracle: Internetpräsenz unter der Domain www.java.com. Online-Quelle unter der URL: <http://www.java.com>, Datum des Zugriffs: 07.07.2017.

Owens (2006)

Owens, M.: The Definitive Guide to SQLite. 1. Auflage, Apress, Berkeley, 2006.

Pankratz (2003)

Pankratz, G.: Zweiseitige kombinatorische Auktionen in elektronischen Transportmärkten – Potenziale und Probleme. Diskussionsbeiträge des Fachbereichs Wirtschaftswissenschaft der FernUniversität in Hagen, Nr. 351, Hagen, 2003.

Papadopoulos/Arbab (1998)

Papadopoulos, G. A.; Arbab, F.: Coordination models and languages. Online-Quelle unter der URL: <http://www.cs.ucy.ac.cy/~george/AdvComp.pdf>, Datum des Zugriffs: 21.05.2013.

Parkes/Ungar (2001)

Parkes, D. C.; Ungar, L. H.: An Auction-Based Method for Decentralized Train Scheduling. In: Proceedings of the Fifth International Conference on Autonomous Agents, 28.05.-01.06.2001 in Montreal, ACM, New York, 2001, S. 43–50.

Parunak et al. (2004)

Parunak, H. V.D.; Brückner, S.; Fleischer, M.; Odell, J.: A design taxonomy of multi-agent interactions. In: Agent-Oriented Software Engineering IV, Springer, Berlin - Heidelberg, 2004, S. 123–137.

Perennes (2013)

Perennes, P.: Use of Combinatorial Auctions in the Railway Industry: Can The “Invisible Hand” Draw The Railway Timetable? Paris, 2013. Auch im Internet verfügbar unter der URL: http://ces.univ-paris1.fr/membre/seminaire/S2I/pdf/18pdf/Perennes_1.pdf, Datum des Zugriffs: 12.04.2013.

Peters (2002)

Peters, R.: Automatisierte Auktionen – Konzeption und Implementierung eines vollautomatischen, multidimensionalen und polypolistischen Marktsystems. In: Wirtschaftsinformatik, Jg. 44 (2002) Nr. 2, S. 131–140.

Pilato et al. (2008)

Pilato, C.; Collins-Sussman, B.; Fitzpatrick, B.: Version Control with Subversion. 2. Auflage, O'Reilly, Sebastopol, 2008.

Porter (2008)

Porter, M. E.: The five competitive forces that shape strategy. In: Harvard Business Review, Jg. 86 (2008) Nr. 1, S. 79–93.

PymProg (2017)

PymProg: Internetpräsenz unter der Domain sourceforge.net. Online-Quelle unter der URL: <http://sourceforge.net/projects/pymprog/>, Datum des Zugriffs: 07.07.2017.

Python (2017)

o.V.: Internetpräsenz unter der Domain www.python.org. Online-Quelle unter der URL: <https://www.python.org>, Datum des Zugriffs: 07.07.2017.

RailcargoOnline (2017)

o.V.: Internetpräsenz unter der Domain www.railcargo-online.com. Online-Quelle unter der URL: <http://www.railcargo-online.com>, Datum des Zugriffs: 07.07.2017.

Rassenti et al. (1982)

Rassenti, S. J.; Smith, V. L.; Bulfin, R. L.: A Combinatorial Auction Mechanism for Airport Time Slot Allocation. In: The Bell Journal of Economics, Jg. 13 (1982) Nr. 2, S. 402–417.

Rebstock (2000)

Rebstock, M.: Elektronische Geschäftsabwicklung, Märkte und Transaktionen – eine methodische Analyse. In: HMD Praxis der Wirtschaftsinformatik, Jg. 37 (2000) Nr. 215, S. 5–15.

Reese et al. (2002)

Reese, G.; Yarger, R.; King, T.; Williams, H.: Managing and Using MySQL. 2. Auflage, O'Reilly, Sebastopol, 2002.

Rothfuss/Ried (2003)

Rothfuss, G.; Ried, C.: Content Management mit XML. 2. Auflage, Springer, Berlin Heidelberg, 2003.

Ruby (2017)

Ruby: Internetpräsenz unter der Domain ruby-lang.org. Online-Quelle unter der URL: <http://ruby-lang.org>, Datum des Zugriffs: 07.07.2017.

RubyGems (2017)

RubyGems: Internetpräsenz unter der Domain rubygems.org. Online-Quelle unter der URL: <http://rubygems.org>, Datum des Zugriffs: 07.07.2017.

RubyOnRails (2017)

RubyOnRails: Internetpräsenz unter der Domain rubyonrails.org. Online-Quelle unter der URL: <http://rubyonrails.org>, Datum des Zugriffs: 07.07.2017.

Sandholm (2002)

Sandholm, T.: Algorithm for optimal winner determination in combinatorial auctions. In: Artificial Intelligence, Jg. 135 (2002), S. 1–54.

Sandholm et al. (2002)

Sandholm, T.; Suri, S.; Gilpin, A.; Levine, D.: Winner Determination in Combinatorial Auction Generalizations. In: Proceedings of the First International Joint Conference on Autonomous Agents and Multi-Agent Systems, 15.-19.07.2002 in Bologna, Bd. 1, ACM, New York, 2002, S. 69–76.

Scheck/Wilske (2011)

Scheck, Ch.; Wilske, S.: Infrastrukturentwicklung im europäischen Korridor Rotterdam-Genua: das INTERREG-IV-B-Projekt Code 24. In: Hege, H.-P.; Knapstein, Y.; Meng, R.; Ruppenthal, K.; Schmitz-Veltin, A.; Zakrzewski, Ph. (Hrsg.): Schneller, öfter, weiter? Perspektiven der Raumentwicklung in der Mobilitätsgesellschaft. 13. Junges Forum der ARL, 13.-15.10.2010 in Mannheim, Verlag der ARL, Hannover, 2011, S. 56–71.

Schmid (1995)

Schmid, U.: Synchronized Universal Time Coordinated for distributed real-time systems. In: Control Engineering Practice, Jg. 26 (1995) Nr. 6, S. 877–884.

Schmidt/Neis (2011)

Schmidt, M.; Neis, P.: OpenStreetMap in der Forschung? In: Anwenderkonferenz für Freie und Open Source Software für Geoinformationssysteme. 05.-07.04.2011 in Heidelberg, 2011, S. 27–29.

Schulze (2012)

Schulze, M.: OpenStreetMap-Daten jetzt mit ODbL. Online-Quelle unter der URL: <http://blog.openstreetmap.org/2012/09/12/openstreertmap-daten-jetzt-mit-odbl/?lang=de>, Datum des Zugriffs: 02.06.2013.

Schwartz et al. (2008)

Schwartz, B.; Zaitsev, P.; Tkachenko, V.; Zawodny, J. D.; Lentz, A.; Balling, D. J.: High Performance MySQL. 2. Auflage, O'Reilly, Sebastopol, 2008.

Schwickert/Pfeiffer (2000)

Schwickert, A. C.; Pfeiffer, E.: Elektronische Marktplätze – Formen, Beteiligte, Zutrittsbarrieren. Arbeitspapiere WI, Nr. 5/2000, Lehrstuhl für Allgemeine BWL und Wirtschaftsinformatik, Johannes Gutenberg-Universität, Mainz, 2000.

Sinatra (2017)

Sinatra: Internetpräsenz unter der Domain sinatrarb.com. Online-Quelle unter der URL: <http://sinatrarb.com>, Datum des Zugriffs: 07.07.2017.

Solis (2012)

Solis, D.: Illustrated C# 2012. Apress, Berkeley, 2012.

Sqlite (2017)

o.V.: Internetpräsenz unter der Domain sqlite.org. Online-Quelle unter der URL: <http://sqlite.org>, Datum des Zugriffs: 07.07.2017.

Stefanov (2010)

Stefanov, S.: JavaScript Patterns. 1. Auflage, O'Reilly, Sebastopol, 2010.

Subversion (2017)

Subversion: Internetpräsenz unter der Domain svn.apache.org. Online-Quelle unter der URL: <http://svn.apache.org>, Datum des Zugriffs: 07.07.2017.

Sudeikat (2010)

Sudeikat, J. O.: Engineering Self-Organizing Dynamics in Distributed Systems: A Systemic Approach. Dissertation, Hamburg, 2010.

Taghavi/Siegel (2007)

Taghavi, M. H.; Siegel, P. H.: Adaptive Methods for Linear Programming Decoding. Online-Quelle unter der URL: <http://arxiv.org/pdf/cs/0703123.pdf>, Datum des Zugriffs: 12.09.2015.

Tate (2005)

Tate, B.: Beyond Java. 1. Auflage, O'Reilly, Sebastopol, 2005.

Tate et al. (2009)

Tate, B.; Carlson, L.; Hibbs, C.: Rails – Up and Running. 2. Auflage, O'Reilly, Sebastopol, 2009.

Ubuntu (2017)

o.V.: Internetpräsenz unter der Domain www.ubuntu.com. Online-Quelle unter der URL: <http://www.ubuntu.com>, Datum des Zugriffs: 07.07.2017.

UIC (2016)

International Union of Railways: NHM – Harmonised Commodity Code – Harmonisiertes Güterverzeichnis – Nomenclature harmonisée des marchandises. Online-Quelle unter der URL: <http://www.uic.org/nhm>, Datum des Zugriffs: 21.08.2016.

United Nations (2001)

o.V.: Terminologie des kombinierten Verkehrs. Prepared by the UN/ECE, the European Conference of Ministers of Transport (ECMT) and the European Commission (EC). United Nations, New York Geneva, 2001. Auch im Internet verfügbar unter der URL: <http://www.oecd.org/sti/transport/roadtransportresearch/1941816.pdf>, Datum des Zugriffs: 19.10.2012.

Van Deursen et al. (2000)

Van Deursen, A.; Klint, P.; Visser, J.: Domain-specific languages: an annotated bibliography. In: ACM Sigplan Notices, Jg. 35 (2000) Nr. 6, S. 26–36.

Van Deursen/Klint (2002)

Van Deursen, A.; Klint, P.: Domain-Specific Language Design Requires Feature Descriptions. In: Journal of Computing and Information Technology, Jg. 10 (2002) Nr. 1, S. 1–17.

W3 (2017)

o.V.: Internetpräsenz unter der Domain www.w3.org. Online-Quelle unter der URL: <http://www.w3.org/standards/xml/core>, Datum des Zugriffs: 07.07.2017.

Wang/Varadharajan (2005)

Wang, Y.; Varadharajan, V.: Trust 2: Developing Trust in Peer-to-Peer Environments. In: Proceedings of the 2005 IEEE International Conference on Services Computing, 11.-15.07.2005 in Orlando, Bd. 1, 2005, S. 24–34.

Watson (2009)

Watson, M.: Scripting Intelligence: Web 3.0 Information Gathering and Processing. 1. Auflage, Apress, Berkeley, 2009.

Weichelt/Bruns (2011)

Weichelt, T.; Bruns, A. S.: Evaluation des Prototyps einer Online-Frachtenbörse für das Verbundprojekt CODE24 – eine kritisch-konstruktive Analyse aus der Perspektive potenzieller betrieblicher Nutzer. Projektbericht des Verbundprojekts CODE24 Nr. 4, Institut für Produktion und Industrielles Informationsmanagement, Universität Duisburg-Essen, Campus Essen, Essen, 2011.

Weyns et al. (2007)

Weyns, D.; Omicini, A.; Odell, J.: Environment as a first class abstraction in multiagent systems. In: Autonomous Agents and Multi-Agent Systems, Jg. 14 (2007) Nr. 1, S. 5–30.

Wieland (2010)

Wieland, B.: Europäische Verkehrspolitik und der Wettbewerb im Eisenbahnwesen und im Straßengüterverkehr. In: Wirtschaftsdienst, Jg. 90 (2010) Nr. 1, S. 43–50.

Wooldrige (1997)

Wooldrige, M.: Agent-Based Software Engineering. In: IEE Proceedings – Software Engineering, Jg. 144 (1997) Nr. 1, S. 26–37.

Wooldrige/Jennings (1995)

Wooldrige, M.; Jennings, N. R.: Intelligent Agents: Theory and Practice. In: The Knowledge Engineering Review, Jg. 10 (1995) Nr. 2, S. 110–152.

Wright/Stevens (1995)

Wright, G. R.; Stevens, W. R.: TCP/IP Illustrated: The Implementation, Bd. 2. 1. Auflage, Addison-Wesley Professional, Boston, 1995.

Yang (2002)

Yang, J.: The Efficiency of an Artificial Double Auction Stock Market with Neural Learning Agents. In: Chen, S.-H. (Hrsg.): Evolutionary Computation in Economics and Finance, 2002, S. 85–105.

Zhang et al. (2011)

Zhang, Y.; Fang, Y.; Wei, K.-K.; Ramsey, E.; McCole, P.; Chen, H.: Repurchase intention in B2C e-commerce – A relationship quality perspective. In: Information and Management, Jg. 48 (2011) Nr. 6, S. 192–200.

Zielstra/Zipf (2010)

Zielstra, D.; Zipf, A.: A Comparative Study of Proprietary Geodata and Volunteered Geographic Information for Germany. In: The 13th AGILE International Conference on Geographic Information Science, 11.-14.05.2010 in Guimarães, Portugal, 2010, o.S. Auch im Internet verfügbar unter der URL: http://agile2010.dsi.uminho.pt/pen/shortpapers_pdf/142_doc.pdf, Datum des Zugriffs: 21.01.2017.

Anhang

A1 – Dokumentation der Datenbanktabellen der IDE

Es folgt die Dokumentation der Datenbanktabellen.

Die Auflistung der Datenbanktabellen dient einer möglichst vollständigen Beschreibung des implementierten Softwareprototyps.

Die folgenden Tabellen zeigen für die jeweiligen Objekte der integrierten Entwicklungsumgebung (IDE), aus welchen Feldern die zugehörige Datenbanktabelle besteht, welchen Datentyp sie ist und einen Beispielwert.

Tabelle 23 zeigt die Beschreibung der in der Spalte Datentyp verwendeten Bezeichner.

Datentyp	Beschreibung	Beispielwert
boolean	Wahrheitswert	true/false
float	Gleitkommazahl	0,5
datetime	Zeitpunkt (inklusive Datum und Zeitverschiebung)	2017-01-01 09:00:00 +0000
integer	Ganzzahl	42
string	alpha-numerische Zeichenkette (max. 255 Zeichen)	„abc123“
text	alpha-numerische Zeichenkette	„Ein beschreibender Text wie dieser.“

Tabelle 23: Beschreibung der Datentypen in der Datenbank¹⁴⁴⁵

Bezeichner	Datentyp	Beschreibung	Beispielwert
color	string	Farbangabe	„#f00“
created_at	datetime	Zeitpunkt der Erstellung	2017-01-01 09:00:00 +0000
host	string	Hostname oder IP-Adresse	„demo.afex-system.org“
name	string	Name	„Anbieter 1“
port	integer	Port	1025
project_id	integer	Referenz auf Projekt	1
updated_at	datetime	Zeitpunkt der letzten Aktualisierung	2017-01-01 09:00:00 +0000
uuid	string	universell eindeutiger Schlüssel	„123e4567-e89b-42d3-a456-426655440000“

Tabelle 24: Beschreibung der Datenbanktabelle „agent_configurations“¹⁴⁴⁶

¹⁴⁴⁵ Quelle: eigene Darstellung.

Bezeichner	Datentyp	Beschreibung	Beispielwert
created_at	datetime	Zeitpunkt der Erstellung	2017-01-01 09:00:00 +0000
entry_id	integer	Referenz auf die eingetragene Agentenkonfiguration	1
owner_id	integer	Referenz auf die eintragende Agentenkonfiguration	2
updated_at	datetime	Zeitpunkt der letzten Aktualisierung	2017-01-01 09:00:00 +0000

Tabelle 25: Beschreibung der Datenbanktabelle „agent_address_book_entries“¹⁴⁴⁷

Bezeichner	Datentyp	Beschreibung	Beispielwert
configuration_id	integer	Referenz auf die eingetragene Agentenkonfiguration	1
scenario_id	integer	Referenz auf das Szenario	2

Tabelle 26: Beschreibung der Datenbanktabelle
„agent_config_simulation_scenarios“¹⁴⁴⁸

Bezeichner	Datentyp	Beschreibung	Beispielwert
created_at	datetime	Zeitpunkt der Erstellung	2017-01-01 09:00:00 +0000
end_at	datetime	Zeitpunkt der spätesten Ankunft	2017-02-02 19:00:00 +0000
end_station_id	integer	Referenz auf den Endpunkt des Transports	2
position	integer	Position im Transportangebot	1
price	float	Preis	1000,00
radius	integer	Radius	20
start_at	datetime	Zeitpunkt der frühesten Abfahrt	2017-02-02 19:00:00 +0000
start_station_id	integer	Referenz auf den Startpunkt des Transports	1
transport_offer_id	integer	Referenz auf das Transportangebot	1
updated_at	datetime	Zeitpunkt der letzten Aktualisierung	2017-01-01 09:00:00 +0000

Tabelle 27: Beschreibung der Datenbanktabelle „agent_transport_offer_sections“¹⁴⁴⁹¹⁴⁴⁶ Quelle: eigene Darstellung.¹⁴⁴⁷ Quelle: eigene Darstellung.¹⁴⁴⁸ Quelle: eigene Darstellung.

Bezeichner	Datentyp	Beschreibung	Beispielwert
configuration_id	integer	Referenz auf die Agentenkonfiguration	1
created_at	datetime	Zeitpunkt der Erstellung	2017-01-01 09:00:00 +0000
flexible	boolean	Angabe, ob Angebot flexiblen Transport abbildet	x
mass	float	Gewicht	50,0
mass	float	Gewicht	50,0
nhms	text	NHM-Nummern	„[‘01000000’]“
updated_at	datetime	Zeitpunkt der letzten Aktualisierung	2017-01-01 09:00:00 +0000
valid_from	datetime	Zeitpunkt der ersten Gültigkeit	2017-01-01 09:00:00 +0000
valid_until	datetime	Zeitpunkt der letzten Gültigkeit	2017-01-01 18:00:00 +0000
volume	float	Volumen	100,0

Tabelle 28: Beschreibung der Datenbanktabelle „agent_transport_offers“¹⁴⁵⁰¹⁴⁴⁹ Quelle: eigene Darstellung.¹⁴⁵⁰ Quelle: eigene Darstellung.

Bezeichner	Datentyp	Beschreibung	Beispielwert
configuration_id	integer	Referenz auf die Agentenkonfiguration	1
created_at	datetime	Zeitpunkt der Erstellung	2017-01-01 09:00:00 +0000
end_at	datetime	Zeitpunkt der spätesten Ankunft	2017-02-01 19:00:00 +0000
end_way_point_id	integer	Referenz auf den Endpunkt des Transports	2
mass	float	Gewicht	50,0
nhms	text	NHM-Nummern	„[‘01000000’]“
price	float	Zahlungsbereitschaft	1000,00
start_at	datetime	Zeitpunkt der frühesten Abfahrt	2017-02-02 19:00:00 +0000
start_way_point_id	integer	Referenz auf den Startpunkt des Transports	1
updated_at	datetime	Zeitpunkt der letzten Aktualisierung	2017-01-01 09:00:00 +0000
valid_from	datetime	Zeitpunkt der ersten Gültigkeit	2017-01-01 09:00:00 +0000
valid_until	datetime	Zeitpunkt der letzten Gültigkeit	2017-01-01 19:00:00 +0000
volume	float	Volumen	100,0

Tabelle 29: Beschreibung der Datenbanktabelle „agent_transport_requests“¹⁴⁵¹¹⁴⁵¹ Quelle: eigene Darstellung.

Bezeichner	Datentyp	Beschreibung	Beispielwert
attempts	integer	Ausführungsversuche	0
created_at	datetime	Zeitpunkt der Erstellung	2017-01-01 09:00:00 +0000
failed_at	datetime	Zeitpunkt des Fehlschlags	2017-01-01 09:01:00 +0000
handler	text	Name des ausführenden Tasks	„PrepareStartParameters“
last_error	text	Fehlerbeschreibung	„error“
locked_at	datetime	Zeitpunkt der Sperrung	2017-01-01 09:01:00 +0000
locked_by	string	Name der sperrenden Entität	„entity1“
priority	integer	Priorität	0
queue	string	Name der Warteschlange	„default“
run_at	datetime	Zeitpunkt der Ausführung	2017-01-01 09:01:00 +0000
updated_at	datetime	Zeitpunkt der letzten Aktualisierung	2017-01-01 09:00:00 +0000

Tabelle 30: Beschreibung der Datenbanktabelle „delayed_jobs“¹⁴⁵²

Bezeichner	Datentyp	Beschreibung	Beispielwert
template	boolean	Vorlage	
created_at	datetime	Zeitpunkt der Erstellung	2017-01-01 09:00:00 +0000
description	text	Beschreibung des Projekts	„Beispielprojekt“
last_opened_at	datetime	Zeitpunkt des letzten Aufrufs	2017-01-02 09:00:00 +0000
name	string	Name des Projekts	„Projekt 1“
start_latitude	float	initialer Breitengrad	51,4502
start_longitude	float	initialer Längengrad	7,01546
start_zoom	integer	initialer Zoom	8
template_icon	string	Vorlagen-Icon	„nrw“
updated_at	datetime	Zeitpunkt der letzten Aktualisierung	2017-01-01 09:00:00 +0000

Tabelle 31: Beschreibung der Datenbanktabelle „projects“¹⁴⁵³¹⁴⁵² Quelle: eigene Darstellung.¹⁴⁵³ Quelle: eigene Darstellung.

Bezeichner	Datentyp	Beschreibung	Beispielwert
action	string	Aktion	„Handshake“
created_at	datetime	Zeitpunkt der Erstellung	2017-01-01 09:00:00 +0000
object_type	string	Art der Metrik	„message“
payload	text	Metadata der Metrik	„{“
receiver_uuid	string	Referenz auf empfangenden Agenten	„123e4567-e89b-42d3-a456-426655440000“
scenario_run_id	integer	Referenz auf Szenarioausführung	1
sender_uuid	string	Referenz auf sendenden Agenten	„e4500670-e45b-43d9-bc23-484687891567“
updated_at	datetime	Zeitpunkt der letzten Aktualisierung	2017-01-01 09:00:00 +0000

Tabelle 32: Beschreibung der Datenbanktabelle „simulation_metrics“¹⁴⁵⁴

Bezeichner	Datentyp	Beschreibung	Beispielwert
created_at	datetime	Zeitpunkt der Erstellung	2017-01-01 09:00:00 +0000
ended_at	datetime	Zeitpunkt der Fertigstellung	2017-01-01 09:01:00 +0000
scenario_id	integer	Referenz auf das Szenario	1
updated_at	datetime	Zeitpunkt der letzten Aktualisierung	2017-01-01 09:00:00 +0000

Tabelle 33: Beschreibung der Datenbanktabelle „simulation_scenario_runs“¹⁴⁵⁵

Bezeichner	Datentyp	Beschreibung	Beispielwert
created_at	datetime	Zeitpunkt der Erstellung	2017-01-01 09:00:00 +0000
name	string	Name des Szenarios	„Szenario 1“
project_id	integer	Referenz auf das Projekt	1
stop_after_auctions	integer	für Abbruch nötige Anzahl an Auktionen	2
updated_at	datetime	Zeitpunkt der letzten Aktualisierung	2017-01-01 09:00:00 +0000

Tabelle 34: Beschreibung der Datenbanktabelle „simulation_scenarios“¹⁴⁵⁶¹⁴⁵⁴ Quelle: eigene Darstellung.¹⁴⁵⁵ Quelle: eigene Darstellung.¹⁴⁵⁶ Quelle: eigene Darstellung.

Bezeichner	Datentyp	Beschreibung	Beispielwert
color	string	Farbangabe	„#f00“
created_at	datetime	Zeitpunkt der Erstellung	2017-01-01 09:00:00 +0000
end_station_id	integer	Referenz auf Endpunkt der Relation	1
kind	string	Art der Relation	„railway“
length_in_km	float	Länge der Relation	12,5
project_id	integer	Referenz auf das Projekt	1
start_station_id	integer	Referenz auf Startpunkt der Relation	2
updated_at	datetime	Zeitpunkt der letzten Aktualisierung	2017-01-01 09:00:00 +0000

Tabelle 35: Beschreibung der Datenbanktabelle „transport_network_relations“¹⁴⁵⁷

Bezeichner	Datentyp	Beschreibung	Beispielwert
created_at	datetime	Zeitpunkt der Erstellung	2017-01-01 09:00:00 +0000
ibnr	string	IBNR des Haltepunkts	„8000098“
kind	string	Art	„Train“
latitude	float	Breitengrad	51,4502
longitude	float	Längengrad	7,01546
name	string	Name des Haltepunkts	„Essen Hbf“
priority	integer	Priorität	8
project_id	integer	Referenz auf das Projekt	1
updated_at	datetime	Zeitpunkt der letzten Aktualisierung	2017-01-01 09:00:00 +0000

Tabelle 36: Beschreibung der Datenbanktabelle „transport_network_stations“¹⁴⁵⁸¹⁴⁵⁷ Quelle: eigene Darstellung.¹⁴⁵⁸ Quelle: eigene Darstellung.

Bezeichner	Datentyp	Beschreibung	Beispielwert
city	string	Stadt	Essen
country_code	string	Ländercode	„de“
created_at	datetime	Zeitpunkt der Erstellung	2017-01-01 09:00:00 +0000
house_number	string	Hausnummer	9
latitude	float	Breitengrad	51,4502
longitude	float	Längengrad	7,01546
postal_code	string	Postleitzahl	„D-45141“
project_id	integer	Referenz auf Projekt	1
search_terms	text	Suchbegriffe	Essen Germany
state	string	Staat/Provinz	North Rhine- Westphalia
station_id	integer	Referenz auf korrespondierenden Haltepunkt	1
street	string	Straße	Universitätsstrasse
suburb	string	Vorort	Mitte
updated_at	datetime	Zeitpunkt der letzten Aktualisierung	2017-01-01 09:00:00 +0000

Tabelle 37: Beschreibung der Datenbanktabelle „transport_network_way_points“¹⁴⁵⁹¹⁴⁵⁹ Quelle: eigene Darstellung.

A2 – Dokumentation des Quelltextes der Software-Agenten

Es folgt die Dokumentation des Quelltextes der protoypisch implementierten Software-Agenten.

Die Dokumentation des Quelltextes dient einer möglichst vollständigen Beschreibung des implementierten Systems autonomer, dezentral organisierter Software-Agenten.

Im Gegensatz zum Haupttext dieser Forschungsarbeit ist die Dokumentation des Quelltextes in englischer Fachsprache verfasst, um einem möglichst breiten Fachpublikum die Arbeit mit dem Quelltext zu ermöglichen.

Zu Beginn wird die README-Datei des Quelltextes wiedergegeben. Diese Datei wird Quelltext-Dokumentationen oftmals vorangestellt, da sie die Voraussetzungen zum Betrieb der Software in technischer Art und Weise erläutert.

README

Installation

The AFEX IDE is a standard Ruby on Rails application.

To install it - alongside the regular AFEX agent software - several prerequisites have to be installed.

Python 2.7

Execute this command to install Python 2.7:

```
$ sudo apt-get install python
```

Ruby

Execute this command to install Ruby:

```
$ sudo apt-get install ruby
```

```
$ # alternatively: build from source  
# https://cache.ruby-lang.org/pub/ruby/2.1/ruby-2.1.10.tar.gz
```

Bundler

Execute this command to install Bundler:

```
$ gem install bundler
```

Redis

Execute these commands to install Redis:

```
$ sudo apt-get install build-essential  
$ sudo apt-get install tcl8.5  
$ wget http://download.redis.io/redis-stable.tar.gz  
$ tar xvzf redis-stable.tar.gz  
$ cd redis-stable  
$ make  
$ sudo make install
```

SQLite

Execute this command to install SQLite:

```
$ sudo apt-get install sqlite3
```

Optional: Git

Execute this command to install Git:

```
$ sudo apt-get install git-core
```

Application setup

Checkout the AFEX repository.

```
$ # a copy of the AFEX repository can be obtained from the author
```

Run bundler inside its working directory.

```
$ cd afex
```

```
$ bundle
```

This will install Ruby on Rails and all other Ruby dependencies of the project.

Next, run `rake db:setup` to setup the database.

```
$ cd web/new-ide
```

```
$ rake db:setup
```

Start the AFEX worker to process background tasks.

```
$ rake jobs:work
```

Start the AFEX IDE from the project's root dir via

```
$ bin/mercury -i
```

Module: Mercury

Defined in: lib/mercury.rb

Overview

Since every software project needs to live in a namespace (which represents its technical and semantical root), the name “Mercury” was chosen as the namespace for this project.

It seemed fitting for a project designing software agents to enable the decentralized trading of transport services, as it refers to the roman god of trade, travels and communication.

Beyond this, the name itself has no further meaning or technical relevance.

Constant Summary

DEFAULT_ENV = 'test'

VERSION = "0.1.0"

Class Attribute Details

(String) log_level

Returns the level for the logger, only messages at that level or higher will be printed.

(String, IO) log_target

Returns the log target, either a filename (**String**) or an **IO** object (typically **STDOUT**).

Class Method Details

(String) env

Returns the environment **Mercury** is started under (defaults to **DEFAULT_ENV**).

Example:

```
Mercury.env  
# => "test"
```

(void) require_all(path = '.')

This method returns an undefined value.

Requires all Ruby files in a given **path**, relative to the **lib** directory.

Example:

```
Mercury.require_all 'mercury/web'
```

(String) root

Returns an absolute path to the root directory of **Mercury**.

Example:

```
Mercury.root  
# => "/home/rf/projects/mercury"
```

Module: Mercury::Action

Defined in: lib/mercury/action.rb

Overview

Action objects are abstract functions which can be performed by an agent.

Each agent runs an “event loop”, which calls a list of **Action** objects sequentially before starting again with the first **Action** object in the list.

These **Action** objects have to observe their environment and react to its state, because they are run over and over again.

Module: Mercury::Action::BaseMethods

Includes: Support::Logging

Included in: CreateMergedGroup, FindBuyers, FindGroupMergeSuggestions,
FindGroupMergers, UpdateGroups

Defined in: lib/mercury/action/base_methods.rb

Overview

This module can be included in `Action` objects to provide access to reader methods for the following properties of the corresponding agent:

- `uuid`,
- `interests`,
- `group_registry`,
- `peer_registry` and
- `comm`.

It also extends the including module with `Mercury::Action::BaseMethods::ClassMethods`.

Class Method Details

(Object) included(other_module)

Called when the module is included into another module.

Module: Mercury::Action::BaseMethods::ClassMethods

Defined in: lib/mercury/action/base_methods.rb

Overview

This module is utilized when the `Mercury::Action::BaseMethods` module is included into another module.

Instance Method Details

(Object) `call(*args)`

Calls the `Action` with the given `args`.

(Object) `call_with(service_lookup, *args)`

Calls the `Action` by looking up the injected dependencies on the given `service_lookup` object. Passes any additional parameters on to `.call` as well.

This is used to conveniently pass an `AgentProxy` object as `service_lookup`.

Class: Mercury::Action::CheckAddressBook

Inherits: Object

Includes: Support::Logging

Defined in: lib/mercury/action/check_address_book.rb

Overview

CheckAddressBook is an Action object that processes all known inactive Peers of an agent and tries to contact them.

Constructor Details

(CheckAddressBook) initialize(address_book, peer_registry, comm)

Returns a new instance of CheckAddressBook.

Instance Attribute Details

(Object) address_book – *readonly*

Returns the value of attribute address_book.

(Object) comm – *readonly*

Returns the value of attribute comm.

(Object) peer_registry – *readonly*

Returns the value of attribute peer_registry.

Class Method Details

(Object) call(*args)

Calls the Action with the given args.

Class: Mercury::Action::CreateMergedGroup

Inherits: Object

Includes: BaseMethods

Defined in: lib/mercury/action/create_merged_group.rb

Overview

CreateMergedGroup is an Action object that merges two groups.

Constructor Details

(CreateMergedGroup) initialize(uuid, interests, group_registry, peer_registry, comm, winner_group, loser_group)

Returns a new instance of CreateMergedGroup.

Instance Attribute Details

(Object) new_group – *readonly*

Returns the value of attribute new_group.

Class: Mercury::Action::FindBuyers

Inherits: Object

Includes: BaseMethods

Defined in: lib/mercury/action/find_buyers.rb

Overview

FindBuyers is an Action object that tries to find buyers for an agents' Interests.

Constructor Details

(FindBuyers) initialize(uuid, interests, group_registry, peer_registry, comm)

Returns a new instance of FindBuyers.

Class: Mercury::Action::FindGroupMergers

Inherits: Object

Extended by: Forwardable

Includes: BaseMethods

Defined in: lib/mercury/action/find_group_mergers.rb

Overview

FindGroupMergers is an Action object that tries to find group's for merging.

Constructor Details

(FindGroupMergers) initialize(uuid, interests, group_registry, peer_registry, comm)

Returns a new instance of FindGroupMergers.

Class: Mercury::Action::FindGroupMergeSuggestions

Inherits: Object

Extended by: Forwardable

Includes: BaseMethods

Defined in: lib/mercury/action/find_group_merge_suggestion.rb

Overview

FindBuyers is an Action object that tries to find group's for merge suggestions.

Constructor Details

(FindGroupMergeSuggestions) initialize(uuid, interests, group_registry, peer_registry, comm)

Returns a new instance of FindGroupMergeSuggestions.

Class Method Details

(Object) call(*args)

Calls the Action with the given args.

Class: Mercury::Action::FindGroupMergeSuggestions::AsBuyer

Inherits: Mercury::Action::FindGroupMergeSuggestions

Defined in: lib/mercury/action/find_group_merge_suggestion.rb

Constructor Details

This class inherits a constructor from Mercury::Action::FindGroupMergeSuggestions.

Instance Method Details

(void) look_for_possible_merge_suggestions

This method returns an undefined value.

Identifies possible recipients for merge suggestions and sends the hosts of those groups a GroupMergeSuggestion message.

Class: Mercury::Action::FindGroupMergeSuggestions::AsSeller

Inherits: Mercury::Action::FindGroupMergeSuggestions

Defined in: lib/mercury/action/find_group_merge_suggestion.rb

Constructor Details

This class inherits a constructor from Mercury::Action::FindGroupMergeSuggestions.

Instance Method Details

(void) look_for_possible_merge_suggestions

This method returns an undefined value.

Identifies possible recipients for merge suggestions and sends the hosts of those groups a GroupMergeSuggestion message.

Class: Mercury::Action::UpdateGroups

Inherits: Object

Extended by: Forwardable

Includes: BaseMethods

Defined in: lib/mercury/action/update_groups.rb

Overview

UpdateGroups is an Action object that processes a given group and updates all its members with its current properties.

Constructor Details

(UpdateGroups) initialize(uuid, interests, group_registry, peer_registry, comm, group = nil)

Returns a new instance of UpdateGroups.

Instance Attribute Details

(Object) group – *readonly*

Returns the value of attribute group.

Class Method Details

(Object) call(*args)

Calls the Action with the given args.

Class: Mercury::Action::UpdateGroups::CompletenessCheck

Inherits: Object

Defined in: lib/mercury/action/update_groups/completeness_check.rb

Overview

Checks for the completeness of a group.

See `#suppliers_complete?` and `#demanders_complete?`

Constructor Details

(CompletenessCheck) initialize(items, all_interests)

Returns a new instance of CompletenessCheck.

Instance Attribute Details

(Object) all_interests – *readonly*

Returns the value of attribute `all_interests`.

(Object) items – *readonly*

Returns the value of attribute `items`.

Instance Method Details

(Boolean) demanders_complete?

Returns `true` if a group's items can be auctioned with the present participants and no member of the group demands any items other than which the group already includes.

(Boolean) suppliers_complete?

Returns **true** if a subset of the group's items can be auctioned with the present participants, i.e. when the group items are demanded by the groups members in a way that some items can be sold inside the group.

Class: Mercury::Action::UpdateGroups:: TickGroup

Inherits: Mercury::Action::UpdateGroups

Defined in: lib/mercury/action/update_groups/tick_group.rb

Overview

TickGroup is the base class for Action objects that process a specific group according to its state.

This is done periodically at a defined interval for each active group in the group registry.

Constructor Details

This class inherits a constructor from Mercury::Action::UpdateGroups.

Instance Method Details

(Boolean) group_demanders_complete?(_group = nil)

Returns **true** if a group's items are all demanded by the group's members and no one is demanding any items besides the offered items.

(Boolean) group_suppliers_complete?(_group = nil)

Returns **true** if a group's items are all demanded by the group's members in a way that all items can be sold inside the group.

Class: Mercury::Action::UpdateGroups:: TickGroupFinal

Inherits: TickGroup

Defined in: lib/mercury/action/update_groups/tick_group/final.rb

Overview

TickGroupFinal is an Action object that processes a given group in the “final” state.

Method Summary

Constructor Details

This class inherits a constructor from Mercury::Action::UpdateGroups.

Class: Mercury::Action::UpdateGroups::TickGroupInAuction

Inherits: TickGroup

Defined in: lib/mercury/action/update_groups/tick_group/in_auction.rb

Overview

TickGroupInAuction is an Action object that processes a given group in the “in_auction” state.

Constructor Details

This class inherits a constructor from Mercury::Action::UpdateGroups.

Instance Method Details

(Object) process

Runs the Action.

Class: Mercury::Action::UpdateGroups:: TickGroupInit

Inherits: TickGroup

Defined in: lib/mercury/action/update_groups/tick_group/init.rb

Overview

TickGroupInit is an Action object that processes a given group in the “init” state.

Constant Summary

TIMER_KEY = :change_to_pre_auction

Constructor Details

This class inherits a constructor from Mercury::Action::UpdateGroups.

Instance Method Details

(Object) process

Runs the Action.

Class: Mercury::Action::UpdateGroups::TickGroupMerged

Inherits: TickGroup

Defined in: lib/mercury/action/update_groups/tick_group/merged.rb

Overview

TickGroupMerged is an Action object that processes a given group in the “merged” state.

Method Summary

Constructor Details

This class inherits a constructor from Mercury::Action::UpdateGroups.

Class: Mercury::Action::UpdateGroups::TickGroupPostAuction

Inherits: TickGroup

Defined in: lib/mercury/action/update_groups/tick_group/post_auction.rb

Overview

TickGroupPostAuction is an Action object that processes a given group in the “post_auction” state.

Constructor Details

This class inherits a constructor from Mercury::Action::UpdateGroups.

Instance Method Details

(Object) process

Runs the Action.

Class: Mercury::Action::UpdateGroups::TickGroupPreAuction

Inherits: TickGroup

Defined in: lib/mercury/action/update_groups/tick_group/pre_auction.rb

Overview

TickGroupPreAuction is an Action object that processes a given group in the “pre_auction” state.

Constant Summary

TIMER_KEY = :change_to_in_auction

Constructor Details

This class inherits a constructor from Mercury::Action::UpdateGroups.

Instance Method Details

(Object) process

Runs the Action.

Class: Mercury::Action::UpdateGroups:: TickGroups

Inherits: Mercury::Action::UpdateGroups

Defined in: lib/mercury/action/update_groups/tick_groups.rb

Overview

TickGroups is an Action object that processes all active groups in the group registry by running a TickGroup object on it that corresponds to its current state.

For a group with the state “init”, a TickGroupInit Action object is created and processed. For a group with the state “pre_auction”, a TickGroupPreAuction is used.

This way, each group is “ticked” every X seconds to process its inner state (where X is determined by the used Config object). The state of individual groups can be changed according to individual circumstances (e.g. a group in the state “init” can be promoted to be a “pre_auction” group based on the group members’ preferences).

Constructor Details

This class inherits a constructor from Mercury::Action::UpdateGroups.

Instance Method Details

(Object) process

Runs the Action.

Class: Mercury::Action::UpdateGroups::UpdateMembers

Inherits: Mercury::Action::UpdateGroups

Defined in: lib/mercury/action/update_groups/update_members.rb

Overview

UpdateMembers is an Action object that processes a given group and updates all its members with its current properties.

Constructor Details

This class inherits a constructor from Mercury::Action::UpdateGroups.

Instance Method Details

(Object) process

Runs the Action.

Module: Mercury::Agent

Defined in: lib/mercury/agent.rb

Overview

The `Agent` module provides fundamental multi-agent system capabilities.

Class: Mercury::Agent::Config

Inherits: Object

Defined in: lib/mercury/agent/config.rb

Overview

Config objects contain configuration for agents to be run.

Constructor Details

(Config) initialize

Returns a new instance of Config.

Instance Attribute Details

(Object) intervals – *readonly*

Returns the value of attribute `intervals`.

(Object) timers – *readonly*

Returns the value of attribute `timers`.

Class Method Details

(Object) for(env, &block)

Returns a Config object for the given `env`.

Class: Mercury::Agent::Config::Intervals

Inherits: Object

Defined in: lib/mercury/agent/config.rb

Overview

Intervals objects contain configuration information about the intervals in which certain Action objects are performed by the agents.

Instance Attribute Details

(Object) check_address_book

Returns the value of attribute check_address_book.

(Object) look_for_buyers

Returns the value of attribute look_for_buyers.

(Object) look_for_mergers

Returns the value of attribute look_for_mergers.

(Object) send_group_updates

Returns the value of attribute send_group_updates.

Class: Mercury::Agent::Config::Timers

Inherits: Object

Defined in: lib/mercury/agent/config.rb

Overview

Timers objects contain configuration information about how long certain **Group** states last before the next state is entered.

Instance Attribute Details

(Object) change_to_in_auction_after

Returns the value of attribute change_to_in_auction_after.

(Object) change_to_pre_auction_after

Returns the value of attribute change_to_pre_auction_after.

Class: Mercury::Agent::Group

Inherits: Object

Includes: Support::Logging

Defined in: lib/mercury/agent/group.rb

Overview

A Group object represents the concept of groups of agents forming to facilitate trade amongst themselves.

Constant Summary

STATE_INIT = 'init'

STATE_PRE_AUCTION = 'pre_auction'

STATE_IN_AUCTION = 'in_auction'

STATE_POST_AUCTION = 'post_auction'

STATE_MERGED = 'merged'

STATE_FINAL = 'final'

STATES = [STATE_INIT, STATE_PRE_AUCTION, STATE_IN_AUCTION,
STATE_POST_AUCTION, STATE_MERGED, STATE_FINAL]

Constructor Details

(Group) initialize(host_uuid, items, uuid = nil)

Returns a new instance of Group.

Instance Attribute Details

(Object) host_uuid – *readonly*

Returns the value of attribute host_uuid.

(Object) items – *readonly*

Returns the value of attribute items.

(Object) `member_uuids` – *readonly*

Returns the value of attribute `member_uuids`.

(Object) `state`

Returns the value of attribute `state`.

(Object) `uuid` – *readonly*

Returns the value of attribute `uuid`.

Instance Method Details

(Boolean) `active?`

Returns `true` if the Group is still active (not merged nor finalized).

(Object) `add_member(uuid)`

Adds a member with the given `uuid` to the Group.

(Object) `auction_accepted!(member_uuid)`

Called when the group's auction was accepted by the member identified via `member_uuid`.

(Boolean) `auction_accepted?`

Returns `true` if all members have accepted the result of the auction.

(Object) `delete_timer(key)`

Delete the timer for the given `key`.

(Boolean) `finalized?`

Returns `true` if the Group is finalized.

(Boolean) host?(peer)

Returns `true` if the given `peer` is the host of the `Group`.

(Boolean) host_or_member?(peer)

Returns `true` if the given `peer` is the host or a member of the `Group`.

(Object) in_state_since

Returns the time (in seconds) that passed since the last state change.

(Boolean) inactive?

Returns `true` if the `Group` is inactive (merged or finalized).

(Object) inspect

Returns an inspected version of the `Group`.

(Boolean) member?(peer)

Returns `true` if the given `peer` is a member of the `Group`.

(Boolean) mergeable?

Returns `true` if the `Group` is mergeable.

(Boolean) merged?

Returns `true` if the `Group` was merged.

(Object) read_timer(key)

Returns the number of seconds that have passed since the timer with the given `key` was started.

(Boolean) selling_any?(demanded_items)

Returns `true`, if the group is selling any of the given `demanded_items`.

(Object) start_timer(key)

Starts a timer for the given key.

(Boolean) timer?(key)

Returns `true` if a timer for the given key exists.

(Object) to_h

Returns a Hash version of the Group.

(Object) update(attributes)

Updates the Group with the given attributes.

Class: Mercury::Agent::GroupRegistry

Inherits: Object

Includes: Support::Logging

Defined in: lib/mercury/agent/group_registry.rb

Overview

The GroupRegistry holds all the groups an agent is participating in.

Constructor Details

(GroupRegistry) initialize

Returns a new instance of GroupRegistry.

Instance Attribute Details

(Object) groups – *readonly*

Returns the value of attribute groups.

Instance Method Details

(Object) [](uuid)

Returns the Group with the given uuid.

(Array<Group>) active_and_merged_groups

Returns all active and merged groups in the registry.

(Array<Group>) active_groups

Returns all active groups in the registry.

(Array<Group>) all

Returns all groups in the registry.

(Object) `create(host_uuid, items, uuid = nil)`

Creates a Group with the given `host_uuid`, `items` and `uuid` (optional).

(Object) `ensure(group_hash)`

Ensures that a group with the properties of the given `group_hash` exists in the GroupRegistry.

(Object) `hosted_by(host)`

Returns all groups hosted by the given `host`.

(Object) `not_hosted_by(host)`

Returns all groups not hosted by the given `host`.

(Object) `register(group)`

Registers a given `group`.

(Object) `selling(_items)`

Returns all groups selling the given `_items`.

Class: Mercury::AgentProxy

Inherits: Object

Includes: Support::Logging

Defined in: lib/mercury/agent_proxy.rb

Overview

An AgentProxy object represents the currently running agent.

It periodically executes Action objects to change its state, which consists of a registry of known peers, known groups and its own preferences in the form of Interests.

Constant Summary

SERVER = Mercury::Communication::Network::Server

CLIENT = Mercury::Communication::Network::Client

Constructor Details

(AgentProxy) initialize(opts)

Returns a new instance of AgentProxy.

Instance Attribute Details

(Object) address_book – *readonly*

Returns the value of attribute address_book.

(Object) group_registry – *readonly*

Returns the value of attribute group_registry.

(Object) interests – *readonly*

Returns the value of attribute interests.

(Object) `peer_registry` – *readonly*

Returns the value of attribute `peer_registry`.

(Object) `server_host` – *readonly*

Returns the value of attribute `server_host`.

(Object) `server_id`

This id is used to stop the server

(Object) `server_port` – *readonly*

Returns the value of attribute `server_port`.

(Object) `uuid` – *readonly*

Returns the value of attribute `uuid`.

Class Method Details

(AgentProxy) `restart(instance)`

Shorthand to restart a given `instance` of an `AgentProxy`.

(AgentProxy) `start(opts)`

Shorthand to start an `AgentProxy` with the given `opts`.

(AgentProxy) `stop(instance)`

Shorthand to stop a given `instance` of an `AgentProxy`.

Instance Method Details

(void) `client_connected(connection)`

This method returns an undefined value.

Called when a client connected from the `AgentProxy`'s server.

(void) client_disconnected(connection)

This method returns an undefined value.

Called when a client disconnected from the AgentProxy's server.

(Object) connect(host, port)

Connects the AgentProxy's server to another agent's server.

(void) connection_to_server_lost(connection)

This method returns an undefined value.

Called when a client has lost a given connection to another agent's server.

(void) connection_to_server_ready(connection)

This method returns an undefined value.

Called when a client has established a given connection to another agent's server.

(Object) inspect

Returns an inspected version of the AgentProxy.

(Object) send_group_update(group)

Sends a GroupUpdate message to each member of the given group.

(void) send_message(receiver_uuid, action, *args)

This method returns an undefined value.

Sends a message using the given action to the given receiver_uuid.

(Boolean) server_running?

Returns true if the server is running.

(void) start_server

This method returns an undefined value.

Starts the server.

(Object) start_timers

Starts all timers.

(void) stop_server

This method returns an undefined value.

Stops the server.

(Object) stop_timers

Stops all timers.

Module: Mercury::Auction

Defined in: lib/mercury/auction.rb

Overview

The Auction module provides capabilities around the preparation and execution of auctions.

Class: Mercury::Auction::Broker

Inherits: Object

Defined in: lib/mercury/auction/broker.rb

Overview

The `Broker` searches through a list of `Peer` objects to find the ones selling any item out of a list of `wanted_items`.

Constructor Details

(Broker) initialize(peers, wanted_items)

Returns a new instance of `Broker`.

Instance Method Details

(Array<Peer>) talk_to

Returns a list of `Peer` objects the agent should talk to.

Class: Mercury::Auction::ExtRunner

Inherits: Object

Defined in: lib/mercury/auction/ext_runner.rb

Overview

The ExtRunner class executes an external script to perform the combinatorial auction algorithm.

It receives back an array consisting of the ids of the winning bids.

Constant Summary

AUCTIONEER_PATH = Mercury.root.join("auctioneer").to_s

DOCKER_CMD = ENV['DOCKER_CMD'] || "docker"

TMP_PATH = File.join(AUCTIONEER_PATH, "tmp")

Constructor Details

(ExtRunner) initialize(bids)

Returns a new instance of ExtRunner.

Instance Attribute Details

(Object) winning_bid_ids – *readonly*

Returns the value of attribute winning_bid_ids.

Class Method Details

(Object) exec(filename)

Executes the auction runner, loading the auction file with the given filename.

Class: Mercury::Auction::Interest

Inherits: Object

Defined in: lib/mercury/auction/interest.rb

Overview

An `Interest` object represents a trading preference of an agent.

It consists of a price, a xor attribute and a list of items.

Constructor Details

(Interest) initialize(options)

Returns a new instance of `Interest`.

Instance Attribute Details

(Object) id – *readonly*

Returns the value of attribute `id`.

(Object) items – *readonly*

Returns the value of attribute `items`.

(Object) price – *readonly*

Returns the value of attribute `price`.

(Object) xor – *readonly*

Returns the value of attribute `xor`.

Instance Method Details

(Boolean) buy?

Returns `true` if the object represents a buying `Interest`.

(Boolean) `sell?`

Returns `true` if the object represents a selling `Interest`.

(Hash) `to_h`

Returns a `Hash` representing the `Interest`.

Class: Mercury::Auction::Interests

Inherits: Object

Extended by: Forwardable

Includes: Enumerable

Defined in: lib/mercury/auction/interests.rb

Overview

An **Interests** (plural) object represents a list of **Interest** (singular) objects.

Constructor Details

(Interests) initialize(arr = nil)

Returns a new instance of **Interests**.

Instance Method Details

(Object) [](key)

Shortcut to read the entry for the given **key**.

(Array<String>) additional_buyers_needed_for(other)

Returns the items for which additional buyers are needed to satisfy the given interests of the other party.

(Object) buying

Returns the buying entries.

(Array) buying_combination_for(items_demanded)

Returns for a given set of **items_demanded** the interests' items that are demanding any subset of these items.

(Boolean) can_sell_to?(other)

Returns `true` if there are any items which can be sold to the other party.

(Array) items_to_sell_to(other)

Returns all items that can be sold to another party.

Example:

Agent 1 sells A+B and C+D. Agent 2 wants to buy A+D, A+B, and D+E+F.

In this case `items_to_sell_to` would return `["A", "D", "B"]`, because these are the items which can be sold to Agent 2.

(void) remove_interests(ids)

This method returns an undefined value.

Removes the interests with the given `ids`. Also removes any interests sharing the same `xor` value as any of the removed interests.

(Object) selling

Returns the selling entries.

(Array) selling_combination_for(items_demanded)

Returns a valid "selling combination" to sell the demanded items in the process.

Example:

Agent 1 wants to sell A+B+C and discovered that he can sell A to somebody. This method returns the valid combination he is offering to effectively sell A (in this case: `["A", "B", "C"]`).

(Array<Hash>) to_a

Returns the entries.

(Object) xored(key)

Returns the entries with the `xor` attribute matching the given `key`.

Module: Mercury::CLI

Defined in: lib/mercury/cli.rb

Overview

The CLI module provides capabilities to run the `./bin/mercury` executable.

Class: Mercury::CLI::OptionsParser

Inherits: Object

Defined in: lib/mercury/cli/options_parser.rb

Constructor Details

(OptionsParser) initialize(args)

Returns a new instance of OptionsParser.

Instance Attribute Details

(Object) args – *readonly*

Returns the value of attribute args.

(Object) options – *readonly*

Returns the value of attribute options.

Instance Method Details

(Object) default_options

Returns the default set of command line options as a Hash.

(Object) parse(&block)

Parses the given command line options from the command line interface.

Class: Mercury::CLI::Runner

Inherits: Object

Defined in: lib/mercury/cli/runner.rb

Constructor Details

(Runner) initialize(args)

Returns a new instance of Runner.

Instance Attribute Details

(Object) opts – *readonly*

Returns the value of attribute opts.

Module: Mercury::Communication

Defined in: `lib/mercury/communication.rb`

Overview

The `Communication` module contains all parts of the software which are concerned with creating messages and network communication.

Class: Mercury::Communication::AddressBook

Inherits: Object

Includes: Support::Logging

Defined in: lib/mercury/communication/address_book.rb

Overview

The AddressBook class is used to keep track of all known contact information.

Constructor Details

(AddressBook) initialize(arr = nil)

Returns a new instance of AddressBook.

Instance Attribute Details

(Array) entries – *readonly*

Returns all known entries.

Instance Method Details

(void) add_address(host, port)

This method returns an undefined value.

Adds an entry to the AddressBook.

(Array) entries_without(peers)

Returns the entries not matching the given peers.

(Boolean) exists?(host, port)

Returns true if an AddressBook entry exists with the given host and port combination.

Class: Mercury::Communication::Message

Inherits: Object

Defined in: lib/mercury/communication/message.rb

Overview

Message objects represent the messages sent between agents to exchange information.

Example:

```
message Handshake {  
  sender "9A01F36D-9ABD-D4D9-80C7-022AF85C82A8"  
  metadata {  
    host "127.0.0.1"  
    port 15001  
    uuid sender  
  }  
}
```

Constructor Details

(Message) initialize(sender_uuid, _attachment = {}, _action = nil)

Returns a new instance of Message.

Instance Attribute Details

(Object) action

Returns the value of attribute action.

(Object) attachment

Returns the value of attribute attachment.

(Object) sender_uuid

Returns the value of attribute sender_uuid.

Class Method Details

(Hash) `decode(string)`

Deserializes the JSON data into a Hash.

(String) `encode(hash)`

Serializes the Message object into JSON.

(Message) `from__data(data)`

Takes raw data and converts it into a Message object.

(Message) `from__payload(payload)`

Takes a payload and converts it into a Message object.

Instance Method Details

(Object) `[](key)`

Shortcut to read metadata for the given key.

(Object) `[]=(key, value)`

Shortcut to write metadata for the given key.

(String) `encoded`

Returns an encoded version of the message.

(Object) `inspect`

Returns an inspected version of the message.

(Hash) `payload`

Returns the message's payload (metadata including attachment data).

Class: Mercury::Communication::MessageHandler

Inherits: Object

Extended by: Forwardable

Includes: Support::Logging

Defined in: lib/mercury/communication/message_handler.rb

Overview

MessageHandler objects define the messages which are sent between agents. They are responsible for composing, formatting and receiving Message objects whose `action` field match their `class_name`.

The following example defines that all FoobarMessage objects contain the `:foo` field of type String.

A FoobarMessage Message object can be created by calling

```
MyApp::FoobarMessage.compose(agent_proxy)
```

```
module MyApp
  class FoobarMessage < Mercury::Communication::MessageHandler
    validate foo: String

    def self.compose(agent_proxy, some_value)
      msg = super(agent_proxy)
      msg[:foo] = "bar!"
      msg[:value] = some_value
      msg
    end

    def receive
      agent_proxy.call_some_method()
    end
  end
end
```


Constructor Details

(MessageHandler) initialize(agent_proxy, message, connection)

Returns a new instance of MessageHandler.

Instance Attribute Details

(Object) agent_proxy – *readonly*

Returns the value of attribute agent_proxy.

(Object) connection – *readonly*

Returns the value of attribute connection.

(Object) message – *readonly*

Returns the value of attribute message.

Class Method Details

(String) action_name

Returns the name of the Action object corresponding to the current MessageHandler class.

Example:

```
module MyApp
  class FoobarMessage < Mercury::Communication::MessageHandler
    validate foo: String
  end
end
```

```
MyApp::FoobarMessage.action_name
# => "Foobar"
```

(Object) compose(agent_proxy, metadata = {})

Called to create a Message object for the given agent_proxy with the given metadata.

(Object) `handle(agent_proxy, message, connection)`

Handles a given `message` for the given `agent_proxy` using the given `connection`.

(String) `handler_class(action_name)`

Returns the `MessageHandler` class corresponding to the given `action_name`.

Example:

```
module MyApp
  class FoobarMessage < Mercury::Communication::MessageHandler
    validate foo: String
  end
end
```

```
MyApp::FoobarMessage.handler_class
# => "Foobar"
```

(Object) `handler_options(action_name)`

Returns the handlers registered under the given `action_name`.

(String) `validate(schema = {})`

Validates the metadata of each incoming message against the provided `schema`.

Example:

```
module MyApp
  class GroupRelatedMessage < Mercury::Communication::MessageHandler
    validate winner_group_uuid: String,
             loser_group_uuid: String
  end
end
```

Instance Method Details

(Array) `errors`

Returns a list of errors resulting from the last validation.

(void) receive

This method returns an undefined value.

Called when a message is received. This is a stub to be overridden by the implementing class.

(Object) set_group_state(group, state)

Sets the **state** property for the given **group**.

(Object) validate(hash)

Validates a received **Message** against the schema defined in the class method **.validate**.

Returns true if the metadata conforms with the defined schema.

Module: Mercury::Communication::MessageHandlers

Defined in: lib/mercury/communication/message_handlers.rb

Overview

The MessageHandlers module contains all messages which agents can use to communicate with one another.

Available messages are

- AuctionAccept
- AuctionResult
- GroupInvite
- GroupMerge
- GroupMergeRequest
- GroupMergeSuggestion
- GroupMergedInvite
- GroupUpdate
- Handshake
- Interests
- JoinGroup
- KeepAlive
- Peers
- RequestPeers

All messages are described in detail in their respective module description.

Class: Mercury::Communication::MessageHandlers::AuctionAccept

Inherits: Mercury::Communication::MessageHandler

Defined in: lib/mercury/communication/message_handlers/auction_accept.rb

Overview

An AuctionAccept message is sent to the host of a group by each member of that group after performing the auction and accepting the resulting allocation.

Example:

```
message AuctionAccept {  
  sender "9A01F36D-9ABD-D4D9-80C7-022AF85C82A8"  
  metadata {  
    uuid sender  
    group_uuid "0b70737e-2bdc-444c-b744-4bb011aff1c8"  
  }  
}
```

Constructor Details

This class inherits a constructor from Mercury::Communication::MessageHandler.

Class Method Details

(Object) compose(agent_proxy, group)

Composes an AuctionAccept message.

Instance Method Details

(Object) receive

Called when an AuctionAccept message is received.

Class: Mercury::Communication::MessageHandlers::AuctionResult

Inherits: Mercury::Communication::MessageHandler

Defined in: lib/mercury/communication/message_handlers/auction_result.rb

Overview

An `AuctionResult` message is sent to each member of a group after performing the auction.

Example:

```
message AuctionResult {  
  sender "9A01F36D-9ABD-D4D9-80C7-022AF85C82A8"  
  metadata {  
    uuid sender  
    group_uuid "0b70737e-2bdc-444c-b744-4bb011aff1c8"  
  }  
}
```

Constructor Details

This class inherits a constructor from Mercury::Communication::MessageHandler.

Class Method Details

(Object) `compose(agent_proxy, group, winning_interest_ids)`

Composes an `AuctionResult` message.

Instance Method Details

(Object) `receive`

Called when an `AuctionResult` message is received.

Class: Mercury::Communication::MessageHandlers::GroupInvite

Inherits: Mercury::Communication::MessageHandler

Includes: ReceiveMethod

Defined in: lib/mercury/communication/message_handlers/group_invite.rb

Overview

A GroupInvite message is sent by the host or a member of a group to agents he wants to join that group.

Example:

```
message GroupInvite {
  sender "9A01F36D-9ABD-D4D9-80C7-022AF85C82A8"
  metadata {
    uuid sender
    group {
      uuid "0d1834f1-ccc5-4610-9667-f8c70ac55536"
      host_uuid "9A01F36D-9ABD-D4D9-80C7-022AF85C82A8"
      items ["..."]
    }
  }
}
```

Constructor Details

This class inherits a constructor from Mercury::Communication::MessageHandler.

Class Method Details

(Object) compose(agent_proxy, group)

Composes a GroupInvite message.

Instance Method Details

(Object) receive

Called when a `GroupInvite` message is received.

Module: Mercury::Communication::MessageHandlers::GroupInvite::ReceiveMethod

Included in: Mercury::Communication::MessageHandlers::GroupInvite,
Mercury::Communication::MessageHandlers::GroupMergedInvite

Defined in: lib/mercury/communication/message_handlers/group_invite.rb

Instance Method Details

(void) receive_group_invite(group_hash)

This method returns an undefined value.

Class: Mercury::Communication::MessageHandlers::GroupMerge

Inherits: Mercury::Communication::MessageHandler

Defined in: lib/mercury/communication/message_handlers/group_merge.rb

Overview

A GroupMerge message is sent to confirm a GroupMergeRequest message. It is received by the host of winner_group.

Example:

```
message GroupMerge {  
  sender "9A01F36D-9ABD-D4D9-80C7-022AF85C82A8"  
  metadata {  
    uuid sender  
    winner_group {  
      uuid "08000c45-59ed-4cf4-8fb1-638d0b80598f"  
      host_uuid "9A01F36D-9ABD-D4D9-80C7-022AF85C82A8"  
      items ["..."]  
    }  
    loser_group {  
      uuid "7261edb5-f80e-4918-81fb-604530ff29c4"  
      host_uuid "9d61e48c-a30d-4eec-9333-b37878b5d3ed"  
      items ["..."]  
    }  
  }  
}
```

Constructor Details

This class inherits a constructor from Mercury::Communication::MessageHandler.

Class Method Details

(Object) compose(agent_proxy, winner_group, loser_group)

Composes a GroupMerge message.

Instance Method Details

(Object) receive

Called when a GroupMerge message is received.

Class: Mercury::Communication::MessageHandlers::GroupMergedInvite

Inherits: Mercury::Communication::MessageHandler

Includes: Mercury::Communication::MessageHandlers::GroupInvite::ReceiveMethod

Defined in: lib/mercury/communication/message_handlers/group_merged_invite.rb

Overview

A GroupInvite message is sent by the host of a group to agents he wants to join that group.

Example:

```
message GroupInvite {  
  sender "9A01F36D-9ABD-D4D9-80C7-022AF85C82A8"  
  metadata {  
    uuid sender  
    winner_group_uuid "08000c45-59ed-4cf4-8fb1-638d0b80598f"  
    loser_group_uuid "7261edb5-f80e-4918-81fb-604530ff29c4"  
  }  
}
```

Constructor Details

This class inherits a constructor from Mercury::Communication::MessageHandler.

Class Method Details

(Object) compose(agent_proxy, group, winner_group, loser_group)

Composes a GroupMergedInvite message.

Instance Method Details

(Object) receive

Called when a GroupMergedInvite message is received.

Class: Mercury::Communication::MessageHandlers::GroupMergeRequest

Inherits: Mercury::Communication::MessageHandler

Defined in: lib/mercury/communication/message_handlers/group_merge_request.rb

Overview

A GroupMergeRequest is sent as an invite to join the sender's winner_group_uuid and give up hosting loser_group_uuid.

Example:

```
message GroupMergeRequest {  
  sender "9A01F36D-9ABD-D4D9-80C7-022AF85C82A8"  
  metadata {  
    uuid sender  
    winner_group_uuid "08000c45-59ed-4cf4-8fb1-638d0b80598f"  
    loser_group_uuid "7261edb5-f80e-4918-81fb-604530ff29c4"  
  }  
}
```

Constructor Details

This class inherits a constructor from Mercury::Communication::MessageHandler.

Class Method Details

(Object) compose(agent_proxy, winner_group, loser_group, initial_request = true)

Composes a GroupMergeRequest message.

Instance Method Details

(Object) receive

Called when a GroupMergeRequest message is received.

Class: Mercury::Communication::MessageHandlers::GroupMergeSuggestion

Inherits: Mercury::Communication::MessageHandler

Defined in:

lib/mercury/communication/message_handlers/group_merge_suggestion.rb

Overview

A GroupMergeSuggestion is sent as a suggestion to merge group1 and group2 sent to both group's hosts.

Example:

```
message GroupMergeSuggestion {
  sender "9A01F36D-9ABD-D4D9-80C7-022AF85C82A8"
  metadata {
    uuid sender
    group1 {
      uuid "08000c45-59ed-4cf4-8fb1-638d0b80598f"
      host_uuid "9A01F36D-9ABD-D4D9-80C7-022AF85C82A8"
      items ["..."]
    }
    group2 {
      uuid "7261edb5-f80e-4918-81fb-604530ff29c4"
      host_uuid "9d61e48c-a30d-4eec-9333-b37878b5d3ed"
      items ["..."]
    }
  }
}
```

Constructor Details

This class inherits a constructor from Mercury::Communication::MessageHandler.

Class Method Details

(Object) compose(agent_proxy, group1, group2)

Composes a GroupMergeSuggestion message.

Instance Method Details

(Object) receive

Called when a `GroupMergeSuggestion` message is received.

Class: Mercury::Communication::MessageHandlers::GroupUpdate

Inherits: Mercury::Communication::MessageHandler

Defined in: lib/mercury/communication/message_handlers/group_update.rb

Overview

A GroupUpdate message is sent periodically to all members of a group with the intention of synchronizing the group's status and information.

Example:

```
message GroupUpdate {  
  sender "9A01F36D-9ABD-D4D9-80C7-022AF85C82A8"  
  metadata {  
    uuid sender  
    group {  
      uuid "08000c45-59ed-4cf4-8fb1-638d0b80598f"  
      host_uuid "9A01F36D-9ABD-D4D9-80C7-022AF85C82A8"  
      items ["..."]  
    }  
  }  
}
```

Constructor Details

This class inherits a constructor from Mercury::Communication::MessageHandler.

Class Method Details

(Object) compose(agent_proxy, group)

Composes a GroupUpdate message.

Instance Method Details

(Object) receive

Called when a GroupUpdate message is received.

Class: Mercury::Communication::MessageHandlers::Handshake

Inherits: Mercury::Communication::MessageHandler

Defined in: lib/mercury/communication/message_handlers/handshake.rb

Overview

A **Handshake** message is sent to introduce the sending agent to another agent.

When a **Handshake** message is received from a previously unknown **Peer**, the agent responds with a **RequestPeers** and an **Interests** message, communicating both its peers and interests to the newly found contact.

Example:

```
message Handshake {  
  sender "9A01F36D-9ABD-D4D9-80C7-022AF85C82A8"  
  metadata {  
    host "127.0.0.1"  
    port 15001  
    uuid sender  
  }  
}
```

Constructor Details

This class inherits a constructor from Mercury::Communication::MessageHandler.

Class Method Details

(Object) compose(agent_proxy, attachment = {})

Composes a **Handshake** message.

Instance Method Details

(Object) receive

Called when a **Handshake** message is received.

Class: Mercury::Communication::MessageHandlers::Interests

Inherits: Mercury::Communication::MessageHandler

Defined in: lib/mercury/communication/message_handlers/interests.rb

Overview

An **Interests** message is sent between agents to exchange interests, i.e. preferences for goods and services to be traded.

Example:

```
message Interests {  
  sender "9A01F36D-9ABD-D4D9-80C7-022AF85C82A8"  
  metadata {  
    uuid sender  
    my_interests ["..."]  
  }  
}
```

Constructor Details

This class inherits a constructor from Mercury::Communication::MessageHandler.

Class Method Details

(Object) compose(agent_proxy, attachment = {})

Composes an **Interests** message.

Instance Method Details

(Object) receive

Called when an **Interests** message is received.

Class: Mercury::Communication::MessageHandlers::JoinGroup

Inherits: Mercury::Communication::MessageHandler

Defined in: lib/mercury/communication/message_handlers/join_group.rb

Overview

A JoinGroup message is sent by an invited agent to the host of a group to join that group.

Example:

```
message JoinGroup {  
  sender "9A01F36D-9ABD-D4D9-80C7-022AF85C82A8"  
  metadata {  
    uuid sender  
    group_uuid "08000c45-59ed-4cf4-8fb1-638d0b80598f"  
  }  
}
```

Constructor Details

This class inherits a constructor from Mercury::Communication::MessageHandler.

Class Method Details

(Object) compose(agent_proxy, group)

Composes a JoinGroup message.

Instance Method Details

(Object) receive

Called when a JoinGroup message is received.

Class: Mercury::Communication::MessageHandlers::KeepAlive

Inherits: Mercury::Communication::MessageHandler

Defined in: lib/mercury/communication/message_handlers/keep_alive.rb

Overview

KeepAlive messages can be utilized by the communication layer to prevent a connection from timing out. Their sole purpose is to keep the connection alive and they have no metadata content.

Example:

```
message KeepAlive {  
  sender "9A01F36D-9ABD-D4D9-80C7-022AF85C82A8"  
}
```

Constructor Details

This class inherits a constructor from Mercury::Communication::MessageHandler.

Instance Method Details

(Object) receive

Called when a KeepAlive message is received.

Class: Mercury::Communication::MessageHandlers::Peers

Inherits: Mercury::Communication::MessageHandler

Includes: ReceiveMethod

Defined in: lib/mercury/communication/message_handlers/peers.rb

Overview

A Peers message is sent to request contact information on other peers.

Example:

```
message Peers {  
  sender "9A01F36D-9ABD-D4D9-80C7-022AF85C82A8"  
  metadata {  
    uuid sender  
    my_peers ["..."]  
  }  
}
```

Constructor Details

This class inherits a constructor from Mercury::Communication::MessageHandler.

Class Method Details

(Object) compose(agent_proxy, attachment = {})

Composes a Peers message.

Instance Method Details

(Object) receive

Called when a Peers message is received.

Module: Mercury::Communication::MessageHandlers::Peers::ReceiveMethod

Included in: Mercury::Communication::MessageHandlers::Peers, RequestPeers

Defined in: lib/mercury/communication/message_handlers/peers.rb

Class Method Details

(Object) included(other)

Called when the module is included into another module.

Instance Method Details

(void) receive_peers(peers)

This method returns an undefined value.

Called when an agent receives the given list of **peers**.

Class: Mercury::Communication::MessageHandlers::RequestPeers

Inherits: Mercury::Communication::MessageHandler

Includes: Peers::ReceiveMethod

Defined in: lib/mercury/communication/message_handlers/request_peers.rb

Overview

A RequestPeers message is sent to request contact information on another agents' known peers. The sending agent sends his own peers to the other agent.

Example:

```
message RequestPeers {  
  sender "9A01F36D-9ABD-D4D9-80C7-022AF85C82A8"  
  metadata {  
    uuid sender  
    my_peers ["..."]  
  }  
}
```

Constructor Details

This class inherits a constructor from Mercury::Communication::MessageHandler.

Class Method Details

(Object) compose(agent_proxy, attachment = {})

Composes a RequestPeers message.

Instance Method Details

(Object) receive

Called when a RequestPeers message is received.

Module: Mercury::Communication::Network

Defined in: lib/mercury/communication/network.rb

Overview

The `Network` module contains all parts of the software which are concerned with network communication.

Class: Mercury::Communication::Network::Client

Inherits: EM::Connection

Includes: SendReceiveData, Support::Logging

Defined in: lib/mercury/communication/network/client.rb

Overview

EventMachine instantiates this class everytime a connection is made

Constant Summary

Constant Summary

Constructor Details

(Client) initialize(host, port, agent_proxy)

Returns a new instance of Client.

Instance Attribute Details

(Mercury::AgentProxy) agent_proxy – *readonly*

(String) host – *readonly*

Returns the host of the Client.

(Fixnum) port – *readonly*

Returns the port of the Client.

Class Method Details

(void) connect(host, port, agent_proxy = nil)

This method returns an undefined value.

Instance Method Details

(Object) post_init

Called when the connection to the server has been established.

(Object) unbind

Called when the connection to the server has been lost.

Module:

Mercury::Communication::Network::MetricsMonitor

Defined in: lib/mercury/communication/network/metrics_monitor.rb

Overview

The MetricsMonitor module provides methods to keep track of agents and their actions from within the IDE.

Constant Summary

SERVER = "localhost"

PORT = ENV["MERCURY_PORT"] || 3000

BASE_URL = "http://#{SERVER}:#{PORT}/simulation/runner/metric"

Class Method Details

(Object) metric_url_auction_done(group_uuid, host_uuid)

(Object) metric_url_group_created(group_uuid, host_uuid)

(Object) metric_url_group_merge(agent, winner_group, loser_group, merged_group)

(Object) metric_url_message(sender_uuid, receiver_uuid, message_type, payload)

(Object) track(&block)

(Object) track_auction_done(group_uuid, host_uuid)

(Object) track_group_created(group_uuid, group_host_uuid)

(Object) track_group_merge(agent, winner_group, loser_group, merged_group)

(Object) track_message(receiver_uuid, message)

Module: Mercury::Communication::Network::SendReceiveData

Included in: Client, Server

Defined in: lib/mercury/communication/network/send_receive_data.rb

Overview

The `SendReceiveData` module contains interface methods for networking. These are the general, not network specific, methods which the client should implement.

Constant Summary

`MESSAGE_DELIMITER =`

Delimiter used to separate message chunks from one another.

`'~' * 23`

Instance Method Details

(DataReceiver) `data_receiver`

Returns the `DataReceiver` used to receive streamed data.

(void) `receive_data(data)`

This method returns an undefined value.

Called by the including `Class` when data is received via a socket connection.

(void) `receive_message(data)`

This method returns an undefined value.

Called by the `DataReceiver` when a full chunk of data has been streamed.

(Object) `send_message(message)`

Sends the given `message`.

Class: Mercury::Communication::Network::SendReceiveData::DataReceiver

Inherits: Object

Defined in: lib/mercury/communication/network/send_receive_data.rb

Overview

The `DataReceiver` class is used to receive data chunks from a socket connection.

Constructor Details

(DataReceiver) initialize(callback_instance, delimiter)

Returns a new instance of `DataReceiver`.

Instance Method Details

(void) receive(data)

This method returns an undefined value.

Called every time a chunk of `data` has been received.

(void) receive_message(data)

This method returns an undefined value.

Called internally when a full chunk of `data` has been streamed.

Class: Mercury::Communication::Network::Server

Inherits: EM::Connection

Extended by: Support::Logging

Includes: SendReceiveData, Support::Logging

Defined in: lib/mercury/communication/network/server.rb

Overview

EventMachine instantiates this class everytime a connection is made.

Constant Summary

Constant Summary

Constructor Details

(Server) initialize(agent_proxy)

Returns a new instance of `Server`.

Instance Attribute Details

(Mercury::AgentProxy) agent_proxy – *readonly*

Class Method Details

(void) start(host, port, agent_proxy = nil)

This method returns an undefined value.

Starts a server for an `agent_proxy` using the given `host` and `port`.

(void) stop(agent_proxy)

This method returns an undefined value.

Stops the server using the given `server_id`.

Instance Method Details

(String) host

Returns the host of the `Server`.

(Fixnum) port

Returns the port of the `Server`.

(Object) post_init

Called when the connection has been established.

(Object) unbind

Called when the connection to a `Client` has been established.

Class: Mercury::Communication::Peer

Inherits: Object

Defined in: lib/mercury/communication/peer.rb

Overview

A Peer object represents another agent in the network including his address and **Interests**.

Constructor Details

(Peer) initialize(uuid, host, port, connection)

Returns a new instance of Peer.

Instance Attribute Details

(EventMachine::Connection) connection

Returns connection for the Peer.

(String) host – *readonly*

Returns the host of the Peer.

(Mercury::Auction::Interests) interests

Returns interests for the Peer.

(Fixnum) port – *readonly*

Returns the port of the Peer.

(String) uuid – *readonly*

Returns unique identifier for the Peer.

Instance Method Details

(Boolean) active?

Returns `true` if the `Peer` has an active connection.

(Object) deactivate!

Deactivates the `Peer`'s connection.

(Object) inspect

Returns an inspected version of the `Group`.

(Hash) to_h

Returns a Hash representing the `Peer`.

(Object) touch

Called when the `Peer` has been active.

Class: Mercury::Communication::PeerRegistry

Inherits: Object

Includes: Support::Logging

Defined in: lib/mercury/communication/peer_registry.rb

Overview

The `PeerRegistry` class is used to store all known `Peers` and keep track of their socket connections, activities and interests.

Constructor Details

(PeerRegistry) initialize

Returns a new instance of `PeerRegistry`.

Instance Attribute Details

(Array) peers – *readonly*

Returns all known `Peers` (including inactive).

Instance Method Details

(Peer) [](uuid)

Returns the found `Peer`.

(Array) active_peers

Returns all active `Peers`.

(Peer) add_peer(uuid, host, port, connection)

Returns the added `Peer`.

(void) close_all_connections

This method returns an undefined value.

Closes all connections.

(Peer) deactivate_peer_by_connection(connection)

Returns the deactivated Peer.

(Peer) find_by_connection(connection)

Returns the found Peer or nil.

(Array<Hash>) to_a

Returns all active Peers as Hash objects.

(Peer) update_peer(uuid, connection)

Returns the updated Peer.

Module: Mercury::DSL

Defined in: lib/mercury/dsl/scoped_dsl.rb

Module: Mercury::DSL::ScopedDSL

Defined in: lib/mercury/dsl/scoped_dsl.rb

Overview

This module provides methods to build a custom DSL:

```
class ProjectScope
  include Mercury::DSL::ScopedDSL

  attribute :name, type: String
  attribute :description, type: String
  alias :desc :description

  attribute :template
  attribute :template_icon, type: String
end
```

```
class ProjectFileScope
  include Mercury::DSL::ScopedDSL

  noop_keyword :project

  builder :Project, ProjectScope
end
```

```
ProjectFileScope.new.eval_file("some_project_file.rb")
```

Class Method Details

(Object) included(other_module)

Called when the module is included into another module.

Instance Method Details

(Object) build_object

This object is returned by the `ScopedDSL::ClassMethods#build` method.

(Object) `eval_code(content, filename = nil)`

Evaluates the given code within this object.

(Object) `eval_file(filename_or_file)`

Reads and evaluates a DSL file.

(Hash) `to_h`

Returns all set instance variables as a Hash. Excludes instance variables with a name starting with a double underscore (e.g. “@__some_variable”)

Example:

```
class UserScope
  include Mercury::DSL::ScopedDSL

  attribute :name, type: String
  attribute :role, type: String
end

user = UserScope.build {
  name    "Alice"
  role    "Admin"
}
user.to_h # => {name: "Alice", role: "Admin"}

user2 = UserScope.build {
  name "Bob"
}
user2.to_h # => {name: "Bob"}
```

Module: Mercury::DSL::ScopedDSL::ClassMethods

Defined in: lib/mercury/dsl/scoped_dsl.rb

Instance Method Details

(Object) attribute(attr_name, options = {})

Specifies a DSL keyword that can be set via a simple method call. The options Hash takes a type parameter to ensure the data-type of the attribute.

Example:

```
class UserScope
  include Mercury::DSL::ScopedDSL

  attribute :name, type: String
  attribute :role, type: String
  attribute :age, type: Integer
  attribute :hobbies, type: Array
end

user = UserScope.build {
  name    "Alice"
  role    "Admin"
  age     23
  hobbies ["Reading", "Hiking"]
}
```

(Object) attributes(*names)

Defines a list of attributes with the given names.

(Object) build(*args, &block)

Builds a DSL object with a block.

Example:

```
class UserScope
  include Mercury::DSL::ScopedDSL

  attribute :name, type: String
```

```
  attribute :role, type: String
  attribute :age, type: Integer
  attribute :hobbies, type: Array
end
```

```
user = UserScope.build {
  name    "Alice"
  role    "Admin"
  age     23
  hobbies ["Reading", "Hiking"]
}
```

(Object) builder(name, builder, options = {})

Specifies a builder for an object inside the DSL.

Example:

```
class UserScope
  include Mercury::DSL::ScopedDSL

  attribute :name, type: String
end
```

```
class UserFileScope
  include Mercury::DSL::ScopedDSL

  builder :User, UserScope
end
```

DSL code:

```
User {
  name "Alice"
}
```

(Object) eval(code = nil, &block)

Evaluates the given code for a new instance of the current class.

(Object) initialize_with(*args)

Specifies instance variables which should be set by the constructor via its arguments. All arguments are then passed to the constructor and saved into instance variables. The names of these instance variables correspond with their position in the call.

Example:

```
class UserScope
  include Mercury::DSL::ScopedDSL

  initialize_with :role, :rank

  attribute :name, type: String
end

user = UserScope.new("TeamCaptain", 1)
user.role # => "TeamCaptain"
user.rank # => 1
```

(Object) noop_keyword(keyword_name)

Specifies a DSL keyword that does not execute any operation, but rather returns the value it is given.

Example:

```
class UserScope
  include Mercury::DSL::ScopedDSL

  attribute :name, type: String
  noop_keyword :admin
end

class UserFileScope
  include Mercury::DSL::ScopedDSL

  builder :User, UserScope
end

# DSL code:
```

```
User {  
  name "Alice"  
  admin true  
}
```

This can be useful if one wants to specify a keyword as part of ones custom DSL, but has not yet figured out its use.

Module: Mercury::Reactor

Defined in: lib/mercury/reactor.rb

Overview

The `Reactor` module provides capabilities to run a reactor loop.

Class Method Details

(Object) `run(&block)`

Note:

This method blocks the calling thread.

It initializes and runs a reactor loop.

The given `block` is executed after initializing the internal event loop but before running it. Therefore this block is the right place to call any code that needs the reactor to run, e.g. starting agents.

Module: Mercury::Support

Defined in: lib/mercury/support/logging.rb

Module: Mercury::Support::Logging

Included in: Action::BaseMethods, Action::CheckAddressBook, Agent::Group, Agent::GroupRegistry, AgentProxy, Communication::AddressBook, Communication::MessageHandler, Communication::Network::Client, Communication::Network::Server, Communication::Network::Server, Communication::PeerRegistry

Defined in: lib/mercury/support/logging.rb

Overview

The `Logging` module adds logging support to any class or module. Once included, it provides a `log` method to log arbitrary messages to the current `log_target`.

The log target can be set via `Mercury.log_target=`.

Example:

```
class SpecialList
  include Mercury::Support::Logging

  def initialize(list)
    log.info "Initializing SpecialList with #{list.inspect} ..."
    # ...
  end
end
```

Instance Method Details

(Logger) `create_logger`

Creates and returns a `Logger`.

(void) `debug_print(message, width = 80, indent = 0)`

This method returns an undefined value.

Prints a `message` to the `Logger` for debugging purposes.

Debugging can be enabled from the command line with the environment variable `DEBUG`.

(Logger) log

Creates a `Logger` if necessary.

Returns the newly created or already present `Logger`.

Module: Mercury::Time

Defined in: lib/mercury/time.rb

Overview

Note:

This interface is also used for testing (i.e. for manipulating the time for all tested agent modules if needed).

The Time module contains methods that provide easy access to the current time and basic time calculations.

Class Method Details

(Fixnum) now

Returns the current timestamp.

Example:

```
Mercury::Time.now  
# => 1427924642
```

(Fixnum) since(timestamp)

Returns the elapsed seconds since the given `timestamp`.

Example:

```
timestamp = Mercury::Time.now  
sleep 5  
Mercury::Time.since(timestamp)  
# => 5
```

Module: Mercury::Web

Defined in: lib/mercury/web.rb

Overview

The Web module contains methods that can start a web server in the same reactor loop as the agents.

Class Method Details

(void) rails(app_dir, agents = nil, host = nil, port = nil)

This method returns an undefined value.

Runs a Rails web app from the web/ subdirectory.

Example:

```
Mercury::Web.rails('ide', start_agents, web_host, web_port)
```

(void) run(options = {})

This method returns an undefined value.

Runs a Rack based app on a webserver (default: thin).

Module: Mercury::Web::AppUtils

Defined in: lib/mercury/web/app_utils.rb

Overview

The AppUtils module is automatically included into a web app (e.g. Rails::Application) started via Mercury::Web.

It adds convenience methods to access the agent proxies running in the same reactor loop as the compatible web app.

Class Method Details

(Object) included(other_module)

Called when the module is included into another module.